Payment Application Data Security Standard (PA-DSS) Implementation Guide

June 3, 2014
Version 5.5.1

The information found in this guide applies to Aptify 5.5.1 and higher.

Information in this document is subject to change without notice. Companies, names and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Aptify®.

The material contained herein is Confidential and Proprietary to Aptify® and protected by non-disclosure provisions in the Aptify Master License Agreements, Aptify Non-Disclosure Agreement, Aptify Partner Agreements, and/or other non-disclosure instruments between the recipient and Aptify. All elements of the material, including but not limited to the content, presentation, and storage and delivery methods are Confidential. In the event that any element of this material is found to not be confidential in a legal proceeding, all other elements will remain Confidential.

Table of Contents

About this Guide

Created / Modified By	Modification Date	Product Change Type	Version	Overview Of Change
Ajaypal Singh	6/3/2014			Updates made for PA-DSS 3.0

This guide is intended for Aptify customers, who want to deploy their Aptify Application which is PA-DSS 3.0 Certified in a PCI DSS (Payment Card Industry Data Security Standard) compliant environment.

PA-DSS Implementation Guide is provided to the customers on secure product implementation, to document the secure configuration specifics mentioned throughout this document, and to clearly delineate vendor, integrator/reseller, and customer responsibilities for meeting PCI DSS requirements. It should detail how the customer and/or integrator/reseller should enable security settings within the customer’s network

The primary purpose of this guide is to Configure Aptify application to run with settings that are compliant with the PCI DSS 3.0 Standard. Following the instructions in this guide in conjunction with those in the relevant user instruction manuals/guides will ensure that Aptify is configured in a PCI-DSS 3.0 compliant manner.

Aptify instructs and advises its customers to deploy Aptify application in a manner that adheres to the PCI DSS v 3.0. Subsequent to this, best practices and hardening methods should be followed and Customer should subscribe to the bulletins such as US-CERT Bulletins, SANS Bulletins, CIS etc. to get latest information on Vulnerabilities & its fixes related to the applications (e.g. IIS Webserver, SQL Server), operating systems to secure their application and environment from the known vulnerabilities.

This guide does not take into account PCI DSS requirements for anything that is not covered by Aptify software. It is understood that the system owner is responsible to implement additional security measures outside the scope of the Aptify System to ensure PCI DSS compliance.

Note: This PA-DSS Implementation Guide is reviewed on a yearly basis, or whenever the underlying application changes or whenever the PA-DSS requirements undergo change. An employee from the Product Management team owns the responsibility to ensure that the PA-DSS Implementation Guide is up to date.

Scope of the Guide

This document is intended for the end users / customers, who use payment and order related functionality of Aptify.

This guide does not take into account PCI DSS requirements for anything that is not classified as Aptify software. It is incumbent on the system owner to implement additional security measures outside the scope of the Aptify system to ensure PCI DSS compliance.

The PCI DSS Security Council has defined a set of 14 best practices, called the Payment Application Data Security Standards (PA-DSS v 3.0), which are aimed at assisting software vendors to create secure payment applications.

About the Application

Aptify family of products includes a Smart Client (Windows Desktop client) used by Aptify customers staff members, and an e-Business application which is web-based and used for member (end-user) access to functionality. Aptify is a CRM application designed for the association market. Aptify customers include organizations like American Bankers Association, American Association of Nurse Anesthetists, and Ohio Society of CPAs. One of the functions of the Aptify product is to process orders for meetings and events, membership subscriptions, periodicals, and merchandise. The product contains functionality to enable staff members and end users to process an order and provide credit cards as payment.

Application Requirements

Core Application	Database Server Requirements	Windows Desktop Application
Aptify 5.5 Aptify 5.5.1	**Server OS:** • Windows Server 2008 (32 or 64-bit) • Windows Server 2008 R2 • Windows Server 2012 (Aptify 5.5.1 and higher) **Server Database Engine:** • SQL Server 2008 • SQL Server 2008 R2 • SQL Server 2012 (Aptify 5.5.1 and higher)	• Windows 8 (Aptify 5.5.1 and higher) • Windows 7 Professional, Enterprise, Ultimate • Windows Vista Business or Ultimate (SP1)
Application Server	**Server Requirements**	
Aptify 5.x.x App Server	• Windows Server 2008 (32 or 64-bit) • Windows Server 2012	
Web Applications	**Server Requirements**	**Supported Client Browsers**
Aptify 5.5 e-Business	• Windows Server 2003 SP2 including Web Edition • Windows Server 2008 (32 or 64-bit) • IIS 6.0 or higher with ASP.NET 2.0.50727 • Aptify 5.5 **Note:** IIS 7.0 supported in 6.0 compatibility mode only.	• Windows: IE7, IE8, IE9, Firefox 1.5+, Chrome Mac OS X: Safari 1.3+, IE5, Firefox 1.5+
Aptify 5.5.1 e-Business	• Windows Server 2008 (32 or 64-bit) (including Web Edition) • Windows Server 2012 (32 or 64-bit) (including Web Edition) • IIS 7.0 or higher with ASP.NET 4.0. 30319 • Aptify 5.5.1	• Windows: IE8, IE9, IE10, Chrome, Firefox 1.5+ • Mac OS X: Safari 1.3+, IE5, Firefox 1.5+
Aptify vNext	• Windows Server 2008 or Windows Server 2008 R2 • Windows Server 2012 • Windows 7 • Microsoft IIS 7.0 or higher with ASP.NET 4.0.30319 • Microsoft .NET Framework 4.5 • Aptify 5.5.1	• Windows: Current versions of Google Chrome, Firefox, IE9, IE10. • Mac OS X: Current version of Safari, Chrome and Firefox. • iOS: Current version of Chrome and Safari. • Android: Current version of Android Browser and Chrome.

Database Server Requirements

This section lists the software and hardware requirements for the Aptify database server. Aptify is a write intensive (disk I/O) application which requires multi-disk RAID arrays to perform adequately. Drive specifications will create more space than is likely necessary to support the Aptify database, but will ensure there are enough disk I/O's to support the system. Note that this document specifies requirements for three levels of hardware support: minimum, mid-point, and high-end.

Database Server

Item	Minimum Requirements	Midpoint Requirements	High-End Requirements
Processor	1 x 2.5 GHz 6-core or higher. Intel E5-2640 or higher, or AMD equivalent	2 x 2.2 GHz 6-core or higher. Intel E5-2660 or higher, or AMD equivalent	2 x 2.9 GHz 6-core or higher. Intel E5-2690 or higher, or AMD equivalent
Storage	Operating System Drive Configuration: 2 x 15K RPM SAS drives in a RAID 1 mirror Example: 2 x 300GB drives = 300 GB of usable space for OS Database Drive Configuration: Minimum of 5 x 15k RPM SAS drives in a RAID 5 array Example: 5 x 300GB drives = 1.1 TB of usable space for the database, logs and tempdb.	Operating System Drive Configuration: 2 x 15K RPM SAS drives in a RAID 1 mirror Example: 2 x 300GB drives = 300 GB of usable space for OS Database Drive Configuration: Minimum of 8 x 15k RPM SAS drives Examples: Medium Transaction Volume RAID 5: 8 x 300GB drives = 2TB of usable space for the database, SQL logs and tempdb High Transaction Volume RAID 1+0: 8 x 300GB drives = 1.2TB of usable space for the database, SQL logs and tempdb	Operating System Drive Configuration: 2 x 15K RPM SAS drives in a RAID 1 mirror Example: 2 x 300GB drives = 300 GB of usable space for OS Database Drive Configuration: Minimum of 14 x 15k RPM SAS drives Example: RAID 1+0: 14 x 300GB drives = 2TB of usable space for the database, SQL logs and tempdb
RAM	12GB at 1333MHz*	24 GB at 1333MHz *	48 GB at 1600MHz or more
Network card	Gigabit Ethernet, 1000 Mbps	Gigabit Ethernet, 1000 Mbps	Gigabit Ethernet, 1000 Mbps

*NOTE: * A 64-bit OS is required to address this memory requirement*

Application Server Requirements

This section lists the software and hardware requirements to support an Application Server for executing asynchronous process flow runs and scheduled tasks. An organization can deploy 1 to *n* application servers, depending on utilization and load.

Software Requirements

- Operating System: Windows Server 2012 or Windows Server 2008
- Aptify 5.5.1 Application Server Component
- Aptify 5.5.1 Windows desktop application
- Microsoft .NET Framework v3.5 SP1: The Aptify setup program adds this to a computer if it is not already installed.

Hardware Requirements

At a minimum, a computer that can run Windows Server 2012 or Windows Server 2008 can also function as an Application Server. However, Aptify recommends that an organization use a computer with at least the following specifications:

- Intel Dual-Core Xeon or Higher
- 4 GB RAM

Windows Desktop Requirements

This section lists the *recommended* software and hardware requirements for running the Aptify Windows desktop application in the supported environments.

Note: The Aptify Windows desktop application will function on hardware that does not meet these requirements, but for best performance, a user's workstation should meet these requirements.

Software Requirements

- Operating System: Windows 8, Windows 7 (Ultimate or Enterprise Editions) or Windows Vista® (Business or Ultimate Edition – SP1)
- Aptify 5.5 Windows desktop application
- Microsoft .NET Framework v3.5 SP1: The Aptify setup program adds this to a desktop computer if it is not already installed.
- Microsoft® Office 2007 or later
  - This includes Office 365 if you are using it in conjunction with the desktop version of Microsoft Office (2007, 2010, or 2013) or if you have Office Professional Plus with Office 365.
  - Microsoft Office is only required if an organization intends to use any of Aptify's Microsoft Office integration features, such as the Export to Excel wizard, the Microsoft Word Mail Merge wizard, Pivot Table views, and Outlook integration.
  - Note that Aptify's Pivot Table view type requires the Office Web Components that are included with Office 2003 (OWC 11).
- Microsoft® MapPoint for Map Views support in Aptify Windows desktop application.

Hardware Requirements

Item	Operating system	Processor	Recommended Hard Drive	Minimum Hard Drive Space	Recommended RAM	Minimum Free RAM required for Application	Network card
Requirements	Windows 8, Windows 7 Pro with SP1 or Vista Business Edition with SP2	Intel Core2 Duo or higher, or AMD equivalent	500GB SATA or 512GB SSD	2GB	4GB	400 MB	100 Mbps

Application Dependencies required to be configured in PCI Compliance Manner

This section lists the application dependencies that are required to configure the application in a PCI Compliance manner. Following this guide, to Configure Aptify application and application dependencies ensure that it runs with settings that are compliant with the PCI DSS 3.0 Standard.

Application Dependencies

- Windows Server 2012 or Windows Server 2008
- SQL Server 2008 or SQL Server 2012
- SQL Server Audit Tool
- Separate Database Server and Web Server
- Microsoft Password Strength Checker
- Memory Cleanup Tool (e.g. Eraser)
- IIS 7 or higher

1 Introduction

Secure payment card processing is an important aspect of responsible business practices, and card processing standards have been developed to ensure that data is securely stored, processed and transmitted. The Payment Card Industry Security Standards Council, LLC (PCI SSC) is the regulatory body that defines the requirements known as the PCI Data Security Standard (PCI-DSS) governing the configuration and secure operations of credit/debit card transactions for an organization.

If you store and process payment card information, you are responsible for ensuring that credit cards are safely handled and stored in compliance with the PCI-DSS standard. If you use a 3rd party application to process credit card transactions, that application must comply with the Payment Application Data Security Standard (PA-DSS), which contains a subset of the PCI-DSS standard and makes complying with PCI-DSS considerably easier.

As of version 5.5.1, Aptify has fulfilled the requirements necessary for a PA-DSS compliant payment application. In order for the merchant to satisfy PA-DSS requirements, it must be deployed as prescribed by Aptify. This document describes the steps that must be followed in order for your Aptify installation to comply with PA-DSS. The information in this document is based on PCI Security Standards Council Payment Application Data Security Standards program (version 3.0, dated November 2013).

1.1 Relationship between PCI-DSS and PA-DSS

The requirements for the *Payment Application Data Security Standard (PA-DSS)* are derived from the *Payment Card Industry Data Security Standard (PCI-DSS) Requirements and Security Assessment Procedures*. The goal of the PA-DSS is to help software vendors develop secure payment applications that do not store prohibited data, such as full magnetic stripe, CVV2 or PIN data, and ensure their payment applications support compliance with the PCI-DSS.

> It is important to note that installing and implementing PA-DSS validated software in a PCI-compliant manner is a necessary, but not sufficient, step in becoming PCI compliant. As a merchant, you must comply with the full PCI-DSS standard.

1.2 Target Audience

This implementation guide is intended for anyone responsible for implementing, maintaining, or administering an Aptify installation. This includes, but is not limited to, Aptify Consulting Services staff, customer IT and administrative staff, resellers, and integrators.

1.3 How to Use This Guide

This guide is arranged in order of the PA-DSS version 3.0 requirements, which must be met by merchants implementing Aptify to become PCI compliant. This document only addresses those specific PA-DSS requirements that merchants must take into account during implementation and in production;

all other PA-DSS requirements are already satisfied by Aptify. For each PA-DSS requirement, the corresponding PCI-DSS requirement is cross-referenced.

> It is recommended that merchants use this guide as a checklist and verify that the all of the guidelines, best practices, and other implementation considerations have been followed. It is only by adhering to all of the recommendations in this document that you can be assured that Aptify is implemented in a PCI-compliant manner.

This guide accompanies the standard documentation set distributed with Aptify software. One of the most important documents is the *Aptify 5.5 Administration Guide*, which provides details on how to setup users and groups as well as entity and field-level security. It is critical that administrators read the *Aptify 5.5 Administration Guide*, the PCI-DSS requirements, and this guide to ensure Aptify is being setup in accordance with PCI standards.

1.4 Resources

There are a number of documents and other sources of information that are important for organizations to consider when planning and implementing PCI-compliance measures. The documents listed below are included in the Aptify software distributions starting with Aptify 5.5.1.

- *PCI Payment Application Data Security Standard - Requirements and Security Assessment Procedures v3.0*. This document is intended to be used to validate that software vendors comply with the PA-DSS. This is the standard Aptify has been validated against.

- *Payment Card Industry (PCI) - Requirements and Security Assessment Procedures v 3.0.* This document contains the requirements against which merchants are evaluated for PCI compliance.

- *Deploying SQL Server 2008 R2 Based on Payment Card Industry Data Security Standards (PCI-DSS) Version 2.0.* The purpose of this white paper is to provide developers and senior technology leaders with technical solutions on how to proactively achieve PCI compliance when deploying SQL Server 2008 R2. This document can also be found at: http://www.parentebeard.com/_docs/PCIwhitepaper-070611.pdf

> Also see the PCI Security Standards Council website's at: www.pcisecuritystandards.org. This website is the official source for all PCI-related information, including standards, questionnaires, merchant and service provider resources, and other reference material.

1.5 Achieving PCI Compliance

It is usually difficult for merchants to achieve PCI compliance on their own given the depth and breadth of the requirements. The PCI Standards Council maintains a list of Qualified Security Assessors (QSAs) who are certified to provide consulting services targeted to merchants working on PCI compliance initiatives. This list is available on the PCI Standards Council website (www.pcisecuritystandards.org).

A very common process for merchants working to achieve PCI compliance is as follows:

1. **Gap Assessment**: The initial step is to engage with a QSA to evaluate the current state of the business. This typically involves on-site interviews with key business and IT personnel regarding handling of cardholder data as well as a technical evaluation of IT security procedures. The result is an assessment document that identifies each compliant and non-compliant requirement in the PCI-DSS standard.

2. **Remediation Phase:** Following the recommendations from the gap assessment, a project to remediate non-compliant requirements is initiated. This could involve putting new policies and procedures in place as well as making changes to IT infrastructure. Many merchants choose to tackle this phase without the use of a QSA, although using outside assistance can accelerate the process.

3. **PCI-DSS Validation:** Once the remediation phase is complete, a QSA can be used to conduct a formal validation against the PCI standards. Getting a formal validation is recommended initially to ensure compliance, although not required except for Level 1 merchants processing over 6 million Visa transactions annually. Your card acquirer will determine your level and compliance requirements.

Beginning with 5.5, Aptify supports credit card processing using PayPal reference transactions. Using this approach, an organization can simplify PCI compliance because Aptify does not store credit card numbers in the database. See “Order Administration” chapter of the *Aptify 5.5 Order Entry Guide* and the” ePayment” chapter of the *Aptify 5.5 Finance and Accounting Financial Integration Guide* (as well as the Aptify 5.5 and 5.5.1 Release Notes) for more information about configuring your system to use reference transactions.

2 Overview of PCI Standards for Merchants

This section describes the key elements of the PCI standards that Aptify customers should be aware of. Visit the PCI Standards Council website (www.pcisecuritystandards.org) for full details.

2.1 PCI Requirements

There are 12 PCI-DSS requirements that are applicable to organizations that store, process and transmit credit or debit card numbers, which are known as Primary Account Numbers (PANs). The table below outlines these 12 requirements. These security requirements apply to all network components, servers and applications that are included in or connected to the cardholder data. Failure to comply with the standards set forth by the PCI-DSS requirements can result in significant fines and higher credit card transaction fees if payment card information is compromised.

There are areas within the PCI-DSS that concern the implementation of Aptify, but the actual standard must be addressed at the company or organization level. If you are processing card-based financial transactions, either through sales and services or as a payment service provider, you must comply with the regulations set forth by the PCI-DSS.

Goals	PCI-DSS Requirements
Build and Maintain a Secure Network	1. Install and maintain a firewall configuration to protect cardholder data 2. Do not use vendor-supplied defaults for system passwords and other security parameters
Protect Cardholder Data	3. Protect stored cardholder data 4. Encrypt transmission of cardholder data across open, public networks
Maintain a Vulnerability Management Program	5. Protect all systems against malware and regularly update anti-virus software or programs 6. Develop and maintain secure systems and applications
Implement Strong Access Control Measures	7. Restrict access to cardholder data by business need-to-know 8. Identify and authenticate access to system components 9. Restrict physical access to cardholder data
Regularly Monitor and Test Networks	10. Track and monitor all access to network resources and cardholder data 11. Regularly test security systems and processes
Maintain an Information Security Policy	12. Maintain a policy that addresses information security for employees and contractors

2.2 Merchant Compliance Levels

As a merchant who processes credit card transactions, you have an obligation to meet all of the requirements in the PCI-DSS regardless of transaction volume. A merchant processing hundreds of transactions annually must meet the same requirements as a merchant processing millions of transactions.

The requirements for validation and reporting of compliance, however, vary by transaction volume. These levels are not governed by the PCI SSC, but by Visa and the other card companies. Your credit card acquirer determines your level and sets your specific requirements based not just on volume, but other factors including prior history of attacks that have resulted in compromised account data.

The table below highlights the key validation requirements for merchants by Visa transaction volume. Other card companies including American Express, Discover, JCB, and MasterCard define their own levels.

Visa Merchant Levels		
Level	**Merchant Criteria**	**Validation Requirements**
1	Merchants processing over 6 million Visa transactions annually (all channels) or Global merchants identified as Level 1 by any Visa region	Annual Report on Compliance ("ROC") by Qualified Security Assessor ("QSA") or Internal Security Assessor ("ISA") if signed by officer of the company Quarterly network scan by Approved Scan Vendor ("ASV") Attestation of Compliance Form
2	Merchants processing 1 million to 6 million Visa transactions annually (all channels)	Annual Self-Assessment Questionnaire ("SAQ") Quarterly network scan by ASV Attestation of Compliance Form
3	Merchants processing 20,000 to 1 million Visa e-commerce transactions annually	Annual SAQ Quarterly network scan by ASV Attestation of Compliance Form
4	Merchants processing less than 20,000 Visa e-commerce transactions annually and all other merchants processing up to 1 million Visa transactions annually	Annual SAQ recommended Quarterly network scan by ASV if applicable Compliance validation requirements set by merchant bank

Source: http://usa.visa.com/merchants/protect-your-business/cisp/merchant-pci-dss-compliance.jsp

2.3 Compensating Controls

An important aspect of the PCI-DSS is the notion of *compensating controls*. Compensating controls may be implemented when an organization cannot meet a requirement explicitly as stated, due to legitimate technical or documented business constraints but has sufficiently mitigated the risk associated with the requirement through implementation of other controls. In order for a compensating control to be accepted, it must meet the following criteria:

1. Meet the intent and rigor of the original stated PCI-DSS requirement.
2. Provide a similar level of defense as the original PCI DSS requirement, such that the compensating controls sufficiently offsets the risk that the original PCI DSS requirement was designed to defend against. (See Navigating PCI DSS for the intent of each PCI DSS requirement.)
3. Be "above and beyond" other PCI-DSS requirements (simply in compliance with other PCI-DSS requirements is not a compensating control).

   When evaluating "above and beyond" for compensating controls, consider the following:

Note: *The items at a) through c) below are intended as examples only. All compensating controls must be reviewed and validated for sufficiency by the assessor who conducts the PCI DSS review. The effectiveness of a compensating control is dependent on the specifics of the environment in which the control is implemented, the surrounding security controls, and the configuration of the control. Companies should be aware that a particular compensating control will not be effective in all environments.*

a) Existing PCI DSS requirements CANNOT be considered as compensating controls if they are already required for the item under review. For example, passwords for non-console administrative access must be sent encrypted to mitigate the risk of intercepting clear-text administrative passwords. An entity cannot use other PCI DSS password requirements (intruder lockout, complex passwords, etc.) to compensate for lack of encrypted passwords, since those other password requirements do not mitigate the risk of interception of clear-text passwords. Also, the other password controls are already PCI DSS requirements for the item under review (passwords).

b) Existing PCI DSS requirements MAY be considered as compensating controls if they are required for another area, but are not required for the item under review. For example, two-factor authentication is a PCI DSS requirement for remote access. Two-factor authentication from within the internal network can also be considered as a compensating control for non-console administrative access when transmission of encrypted passwords cannot be supported. Two-factor authentication may be an acceptable compensating control if: (1) it meets the intent of the original requirement by addressing the risk of intercepting clear-text administrative passwords; and (2) it is set up properly and in a secure environment.

c) Existing PCI DSS requirements may be combined with new controls to become a compensating control. For example, if a company is unable to render cardholder data unreadable per Requirement 3.4 (for example, by encryption), a compensating control could consist of a device or combination of devices, applications, and controls that address all of the following: (1) internal network segmentation; (2) IP address or MAC address filtering; and (3) two-factor authentication from within the internal network.

4. Be commensurate with the additional risk imposed by not adhering to the PCI-DSS requirement.

The assessor is required to thoroughly evaluate compensating controls during each annual PCI DSS assessment to validate that each compensating control adequately addresses the risk the original PCI DSS requirement was designed to address, per items 1-4 above. To maintain compliance, processes and controls must be in place to ensure compensating controls remain effective after the assessment is complete.

3 Aptify Guidelines for Adhering to PA-DSS Requirements

3.1 Overview of Aptify Protection of Credit Card Data

Aptify software has been developed with the goal of protecting cardholder data in accordance to the PA-DSS standards. Some of these protections are built directly into the software, and others require implementation and system management consideration once the system is live.

The following is a summary of the key areas that the core Aptify software addresses, specifically as it relates to the PA-DSS and the PCI-DSS compliance efforts of our customers.

- For cardholder data, Aptify stores only the account holder name, the PAN (Primary Account Numbers) and expiration date. Aptify does not store any other data, including card validation code (CVV, CVV2, CVC2, CID, CAV2, etc.), PIN, or magnetic stripe data.

- The PAN is stored in a single table in the Aptify database, and is encrypted using AES 256-bit encryption.

- Within Aptify, the display of the PAN is always masked and never displayed in full on any screen or report. It is also never written to any history, log, or diagnostic files. Details are provided later in this document regarding the database table locations and procedures to safely purge encrypted PANs from the system. See “Using the Credit Card Purge Utility” for more information.

- Aptify supports split-key management techniques that enable IT organizations to restrict access to users and system administrators from having access to all of the information required to obtain PANs, including the encrypted card data, the encryption keys, and the keys that encrypt the encryption keys. See “Key Management and Protection of Keys” for more details.

3.2 Secure Deletion of Sensitive Data from Older Versions

PA-DSS requirement: 1.1.4, 1.1.5
PCI-DSS requirement: 3.2

This requirement states that vendors must provide a mechanism to securely delete any sensitive data stored by prior versions of the software, including magnetic stripe data, Card Validation Code (CVV), PINs, etc. Current and prior versions of Aptify do not and have never stored or written this data in the database or other locations, so these requirements are not applicable.

> In accordance with PA-DSS requirement 1.1.5, customers are strongly encouraged not to collect, store, or write additional sensitive information unless absolutely necessary for diagnostic purposes. If this information is collected, the information should be securely deleted immediately after use using secure deletion tools, such as Eraser (http://eraser.heidi.ie/).

3.3 Cardholder Data Retention Policy

PA-DSS requirement: 2.1
PCI-DSS requirement: 3.1

Customers are required to develop a cardholder data retention and disposal policy to purge cardholder data after expiration of a customer-defined retention period.

Aptify supports this requirement by providing a utility that enables customers to purge cardholder data that meets customer-defined criteria. In addition, information is also provided below regarding the specific database locations that contain cardholder data so customers can write their own scripts.

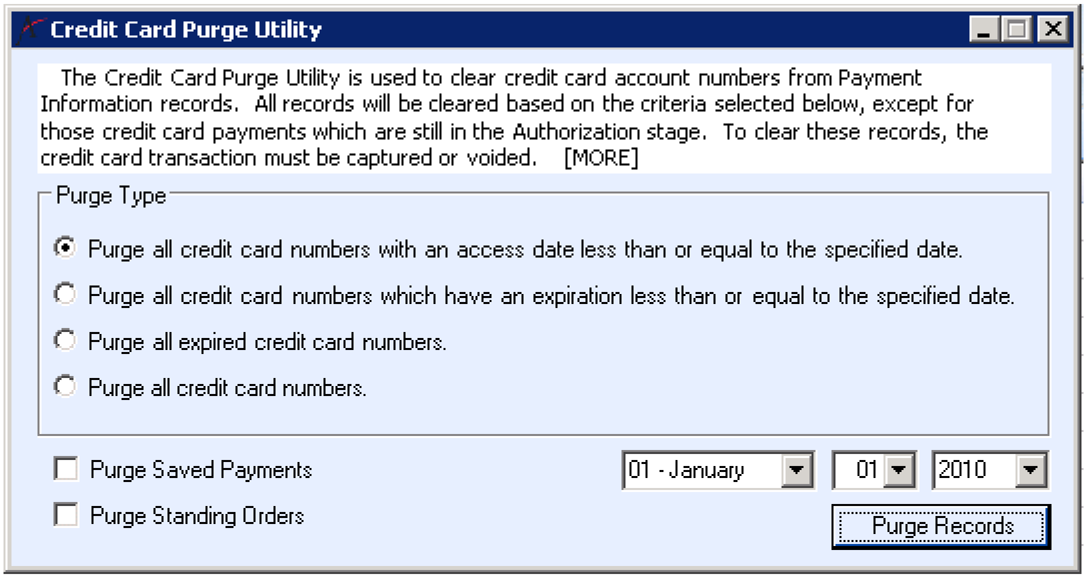
> Beginning with 5.5, Aptify supports credit card processing using PayPal reference transactions. Using this approach, an organization can simplify PCI compliance because Aptify does not store credit card numbers in the database. See “Order Administration” chapter of the *Aptify 5.5 Order Entry Guide* and the” ePayment” chapter of the *Aptify 5.5 Finance and Accounting Financial Integration Guide* (as well as the Aptify 5.5 and 5.5.1 Release Notes) for more information about configuring your system to use reference transactions.

3.3.1 Using the Credit Card Purge Utility

All credit card numbers are stored within the PaymentInformation table within the Aptify database. To purge credit card numbers from Aptify, a DBA would only need to target this table to clear this data. Removing the data within this table may affect your ability to further process the credit card payment, so it is important that all payments have been captured or voided prior to the removal of the credit card number.

Aptify provides a Credit Card Purge Utility for a convenient way to handle this task. It will allow administrative users to remove credit card account numbers based on a specific criterion. The partial credit card number, which can only be used for reference purposes, will not be removed by this utility.

> By default, this utility is only available to administrative users of Aptify, and it is highly recommended that a secure backup is made prior to running this utility and that a memory cleanup tool, such as Eraser (http://eraser.heidi.ie/) is used after running the tool.



The Credit Card Purge Utility is used to clear credit card account numbers from Payment Information records. All records will be cleared based on the criteria selected below, except for those credit card payments which are still in the Authorization stage. To clear these records, the credit card transaction must be captured or voided. [MORE]

Purge Type

● Purge all credit card numbers with an access date less than or equal to the specified date.

○ Purge all credit card numbers which have an expiration less than or equal to the specified date.

○ Purge all expired credit card numbers.

○ Purge all credit card numbers.

□ Purge Saved Payments

□ Purge Standing Orders

01 - January | 01 | 2010

Figure 1 – The Credit Card Purge Utility

The Credit Card Purge Utility includes the following from top to bottom (and left to right):

- **Tool Description and Help:** This section includes a summary of the tool and available options. Selecting the **[MORE]** option will display additional help information for the tool.

- **Purge Type Options:** Determines the type of purge to perform. See the “Purge Options” below for more details.

- **Purge Saved Payments:** Optional selection to include Saved Payment Methods records for both persons and companies. See the “Purge Options” below for more details.

- **Purge Standing Orders:** Optional selection to include Standing Orders records. See the “Purge Options” below for more details.

- **Date Control:** The Month, Day and Year Date Control drop-down boxes are available when applicable, depending on the purge type selected.

- **Purge Records:** After an administrator has determined the proper purge criteria, selecting this button, begins the purging process.

3.3.1.1 Purge Options

There are four choices for removing credit card numbers:

1. Purge all credit card numbers with an access date less than or equal to the specified date.
2. Purge all credit card numbers which have an expiration less than or equal to the specified date.
3. Purge all expired credit card numbers.
4. Purge all credit card numbers.

Optionally, you may include saved credit card payments and standing orders for companies or persons with all removal types. Below is a more detailed description of each option:

Note Concerning Authorized and Refund Payments: Payment Information records which are still marked as 'Authorized', *with the exception of refund payments*, cannot be purged. These records must be captured or voided to be cleansed. Refund payments, though marks as 'Authorized' are cleansed.

Option 1: Purge all credit card numbers with an access date less than or equal to the specified date.

The first option will delete all card numbers which have not been used or accessed since a date specified by the user of the utility. In Aptify 5.0 SP2, all Payment information records have a date updated field which records the last time the payment information record was modified or the card was used. The specific conditions for when this field is updated are:

1. The record is updated whenever the Payment Information record is modified, for example, when updating the credit card number or expiration date.

2. The record is updated whenever the credit card is used in a transaction through the e-Payments module or from a standing order.

3. The record is not updated if an order is processed using a Saved Payment Method. However, the Purge Utility will not delete credit card numbers from any Payment Information records that are referenced by an active saved payment method with an End Date greater or equal to today (the date the utility is run). If the "Purge Saved Payments" and "Purge Standing Orders" options are checked, the expiration date will be determined as the last access date. System administrators may choose to expire additional saved payment methods/standing orders (or set them to inactive), if they wish for those card numbers to be candidates for deletion when the optional saved payment/standing order options are checked.

The utility will compare the date specified in the purge tool with the updated date field in Aptify, and delete any records with a Date Updated date equal to or less than the date specified.

Option 2: Purge all credit card numbers which have an expiration less than or equal to the specified date.

The second option will delete all records that contain credit card numbers with expiration dates which are equal to or less than the month and year provided by the user. This would provide the user the ability to delete all credit card numbers for cards that have expired several months ago, but to keep recently expired ones. If the "Purged Saved Payments" option is selected, the credit card numbers within a person's Saved Payment Methods records and a company's Saved Payment Methods records will also be checked if it is not marked as active or if it is past the active date for that record (determined by the payment method's End Date field). If the "Purged Standing Orders" option is selected, the credit card numbers associated with standing orders will also be checked if the Standing Orders record has a Status of Inactive or if the Standing Orders record is past the active date for that record (determined by the standing order's Date Expires field).

Option 3: Purge all expired credit card numbers.

The third option will allow the user to purge all expired credit card numbers from the payments. All credit card numbers with expiration dates of less than this month will be removed. If the "Purged Saved Payments" option is selected, expired cards linked to persons and companies Saved Payment Methods records, which are not active or have an end date less than the present day, will also be expunged. If the "Purge Standing Orders" option is selected, expired cards linked to inactive standing orders or standing orders that have an expired date less than the present day, will also be expunged.

Option 4: Purge all credit card numbers.

The last option will delete all payment credit card numbers from the Payment Information table not marked as 'Authorized' and all save payment/standing orders records which are not active (i.e. marked as inactive and having an end date/expired date in the past) automatically without having to select the Purge Saved Payments/Purge Standing Orders options. With this purge type, selecting the Purge Saved Payments and/or Purge Standing Orders options clears the credit card account numbers of *all active* saved payments and/or standing orders as well.

3.3.1.2 Running the Credit Card Purge Tool

Follow the steps below to run Aptify's Credit Card Purge Utility:

> It is highly recommended that a secure backup is made prior to running this utility and that a memory cleanup tool, such as Eraser (http://eraser.heidi.ie/) is used after running the tool.
>
> Also note that Payment Information records which are still marked as 'Authorized' cannot be purged. These records must be captured or voided to be cleansed.

1. Launch the **Credit Card Purge Tool** from the Payments service.

- The purge utility is associated with the Payments service within Aptify's Order Entry application and is available for users belonging to the administrators group.

- The **Credit Card Purge Tool** icon appears in the service toolbar or view toolbar for the Payments service.

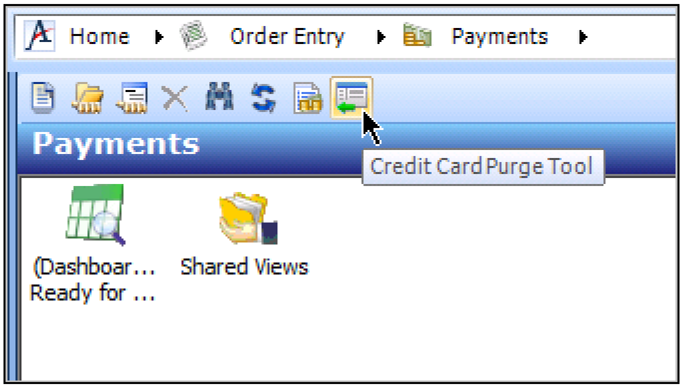
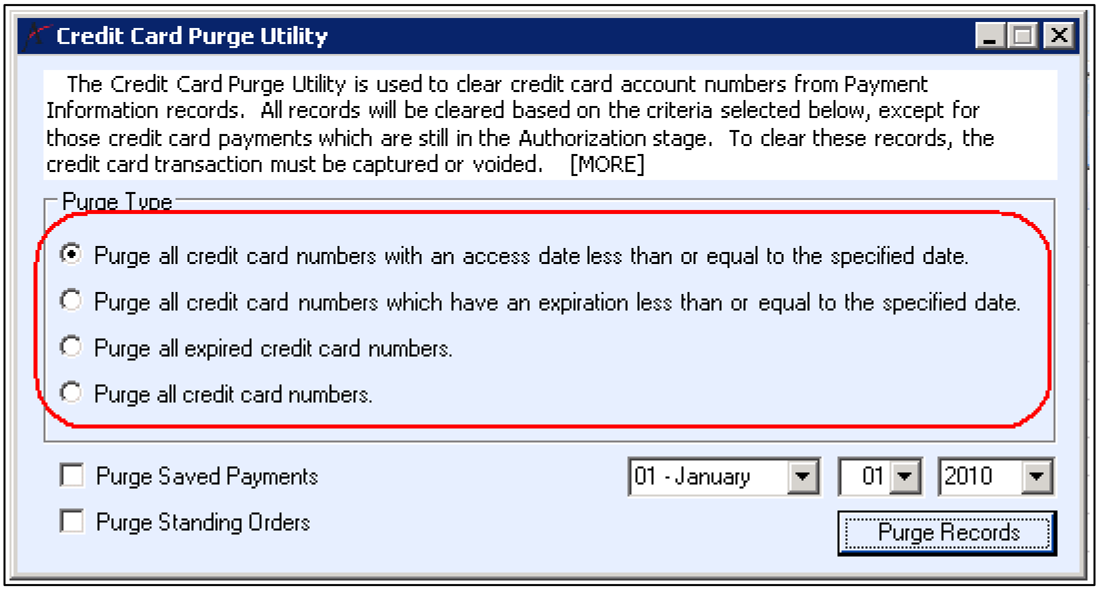


Figure 2 - Credit Card Purge Tool Button

2. Select the appropriate **Purge Type** option based on the desired removal criteria.

   - See the "Purge Options" for more details about the available options.

   - Selecting the **[MORE]** option in the description section of the utility will display additional help information for the tool.



Credit Card Purge Utility

The Credit Card Purge Utility is used to clear credit card account numbers from Payment Information records. All records will be cleared based on the criteria selected below, except for those credit card payments which are still in the Authorization stage. To clear these records, the credit card transaction must be captured or voided. [MORE]

Purge Type

● Purge all credit card numbers with an access date less than or equal to the specified date.
○ Purge all credit card numbers which have an expiration less than or equal to the specified date.
○ Purge all expired credit card numbers.
○ Purge all credit card numbers.

□ Purge Saved Payments
□ Purge Standing Orders

01 - January | 01 | 2010

Figure 3 - Purge Utility Options

3. If you also want to purge Saved Payment Methods record, select the **Purge Saved Payments** option.

   - Note that the "Purge all credit card numbers." option (the last option) deletes credit card numbers associated with inactive saved payments automatically without having to select the saved payments option. With this purge type, selecting the Purge Saved Payments option clears the credit card account numbers of *all active* saved payments as well.
   - See the "Purge Options" on 21 for more information about purging credit card numbers associated with saved payment methods, depending on the purge type specified.

4. If you also want to purge the credit card numbers associated with Standing Orders, select the **Purge Standing Orders** option.

   - Note that the "Purge all credit card numbers." option (the last option) deletes credit card numbers associated with expired Standing Orders automatically without having to select the standing orders option. With this purge type, selecting the Purge Standing option clears the credit card account numbers of *all active* standing orders as well.

   - See the "Purge Options" for more details about purging credit card numbers associated with standing orders, depending on the purge type specified.

5. Select the appropriate **Month**, **Day** and **Year** (when applicable) in the date control fields.

   - The Month and Year date controls are only available for the first two purge types and the Day date control is only available for the first purge type.

6. Once the appropriate purge options have been selected and the appropriate date specified, select the **Purge Records** button to begin the purge process.

3.3.2 Credit Card Information in the Database

The only location where the Credit Card numbers and their corresponding expiration dates are stored is within the Payment Information table within the Aptify database. Aptify should never be configured to store Cardholder data in a location other than in this table. Attempts to store Cardholder data in any other table within Aptify may violate the terms of your software license and eliminate all software warranties.

The PCI-relevant columns in the Payment Information table are:

Column	Purpose
PaymentInformation.CCAccountNumber	Encrypted credit card number
PaymentInformation.CCExpire	Credit card expiration date
PaymentInformation.CCPartial	Masked credit card number

DateUpdated	The date the payment information record was last modified, which includes the date any payment made using this payment record was last processed.

Also, note that PANs encrypted using a SQL Server Encryption key (as explained in section 3.6.2 of this document) will be unrecoverable if a database backup is restored to a different SQL instance (since the encryption key is tied to a specific SQL Server instance). However, if you create database backups for purposes other than system disaster recovery, Aptify strongly recommends that you remove all encrypted PANs from the system when creating the backup. Aptify provides a sanitization script for this purpose. See the Aptify Database Sanitization Tools and the “Creating a Sanitized Aptify Database Backup” document for details.

3.3.3 Related Tables

A number of other Aptify tables contain foreign key references back to the PaymentInformation table. There is no need to remove records that link to the PaymentInformation table; these are provided for reference only.

- BulkOrderCustomer
- BulkWriteOff
- CashCtrlBatchDetail
- ClassRegistration
- CompanySavedPaymentMethod
- GrantReportPaymentLink
- HousingReservationGuest
- MembershipEnrollment
- OrderMaster
- Payment
- PaymentAuthorization
- PaymentDetail
- PaymentGLEntry
- PersonSavedPaymentMethod
- StandingOrder
- StandingOrderSchedule

3.4 Mask PAN When Displayed

PA-DSS requirement: 2.2
PCI-DSS requirement: 3.3

The PA-DSS standard states that the PAN must be masked where ever displayed, unless there is a legitimate business justification. Aptify supports this requirement out-of-the-box by masking the PAN whenever card information should be displayed. The standard Aptify product does not contain any instances of unencrypted PANs.

Aptify strongly recommends against tampering with the system in an attempt to display unencrypted PANs, and such actions may violate the terms of your software license and eliminate all software warranties. Please note that it is up to the customer to ensure appropriate controls are in place to mitigate risk such that PCI compliance can be achieved.

3.5 Render PAN unreadable anywhere it is stored

PA-DSS requirement: 2.3
PCI-DSS requirement: 3.4

The PA-DSS standard states to render PAN unreadable anywhere it is stored (including data on portable digital media, backup media, and in logs). Aptify ensures that this requirement is met by storing the PAN in a single table in the Aptify database, and encrypting it using AES 256-bit encryption.

3.6 Key Management and Protection of Keys

PA-DSS requirement: 2.4, 2.5, 2.6
PCI-DSS requirement: 3.5, 3.6

Payment applications must protect cryptographic keys used for the encryption of cardholder data against disclosure and misuse. Aptify supports these requirements by using strong encryption (AES 256-bit) for all keys. Aptify supports encryption of the PAN, plus encryption of the key that encrypts the PAN. The location of the Encryption key is located in the Aptify DB. The Key Encryption Key can be located in the master DB, or as a separate file containing the decryption key (key encryption key). The encryption uses AES (Rijndael) algorithm, but the actual key used would be subject to the same rules as any password. You can test the strength of the key/password using this Microsoft's Password Strength Checker or any other such software.

The payment application must implement key management techniques for keys used for the encryption of cardholder data. There are several requirements that must be supported (and that Aptify does support), including:

- The payment application must support "dual-control" of keys by separating the PAN encryption keys from the key encryption keys such that an IT organization can limit exposure of IT personnel who may have all information necessary to decrypt credit card data.
- The payment application must support key rotation, retiring of old keys, and re-encrypting PANs in accordance with PCI guidelines.
- Assurance that substitution of keys will render the data unusable.

Basic Key Provider, SQL Encryption Key Provider and File Key Provider are the three Key Providers that can be used for Security Keys. Aptify recommends the usage of SQL Encryption Key Provider, using which cryptographic keys are securely stored and the details are available in Section 3.6.2. PAN Encryption Key is stored in Aptify Database table named SecurityKey, but the key value stored is encrypted using the certificate generated by the Database Master Key. It can be verified that keys are securely stored by looking at the database table, where you can only see the encrypted value of the PAN Encryption Key.

However, if the security key provider being used is the Basic Key Provider, you will need to secure the key, which is retrievable from the database – if the database is ever out of the control of the client. In order to prevent the retrieval of the key when using the Basic Key Provider, the encryption keys used to encrypt the PAN should be encrypted using database level encryption such as SQL Server's Symmetric Key Encryption. The Key Value column of the Security Key should be encrypted using symmetric encryption capabilities found within SQL Server (http://technet.microsoft.com/en-us/library/ms179331(v=sql.105).aspx). By implementing this method, it will be impossible for a hacker to extract credit cards from a database backup or by accessing the database without being the system administrator.

Aptify does not recommend distribution of cryptographic keys. In case, distribution of cryptographic keys is absolutely necessary, you can store the key on a text file and zip that text file with a strong password. Also make sure that it is not stored on the same system preferably not on the same network. For additional security it can be stored on a physical key (Pen Drive).

3.6.1 Dual Control, Separation of Duties and Split Knowledge

According to PCI requirements it is required that encryption keys be stored separately from the data they protect, and to make sure that the people who manage encryption keys are not the people who manage the protected data. Aptify recommends using the concept of Dual Control, Separation of Duties and Split Knowledge for better Key Management.

Dual Control means that no one person should be able to manage your encryption keys. Creating and defining access controls should require at least two individuals working together to accomplish the task. This can be achieved by having an account which has a password, whose first half is known to one person and second half is known to another person or using some similar methodology.

Separation of Duties means that different people should control different aspects of your key management strategy. This is the old adage, "Don't put your eggs in one basket." People who create and manage the keys should not have access to the data that they protect. And, the person with access to protected data should not be able to manage encryption keys.

Split Knowledge applies to the manual generation and substitution of encryption keys. More than one person should be required to constitute or re-constitute a key in this situation. The only time cryptographic key can be seen in plain text is while entering the value in the new Key Security Key record, before saving the record.

3.6.2 Implementing Key Encryption using SQL Server

SQL Server uses encryption keys to help secure data, credentials, and connection information that is stored in a server database. SQL Server has two kinds of keys: symmetric and asymmetric. Symmetric keys use the same password to encrypt and decrypt data. Asymmetric keys use one password to encrypt data (called the public key) and another to decrypt data (called the private key).

In SQL Server, encryption keys include a combination of public, private, and symmetric keys that are used to protect sensitive data. The symmetric key is created during SQL Server initialization when you first start the SQL Server instance. The key is used by SQL Server to encrypt sensitive data that is stored in SQL Server. Public and private keys are created by the operating system and they are used to protect the symmetric key. A public and private key pair is created for each SQL Server instance that stores sensitive data in a database.

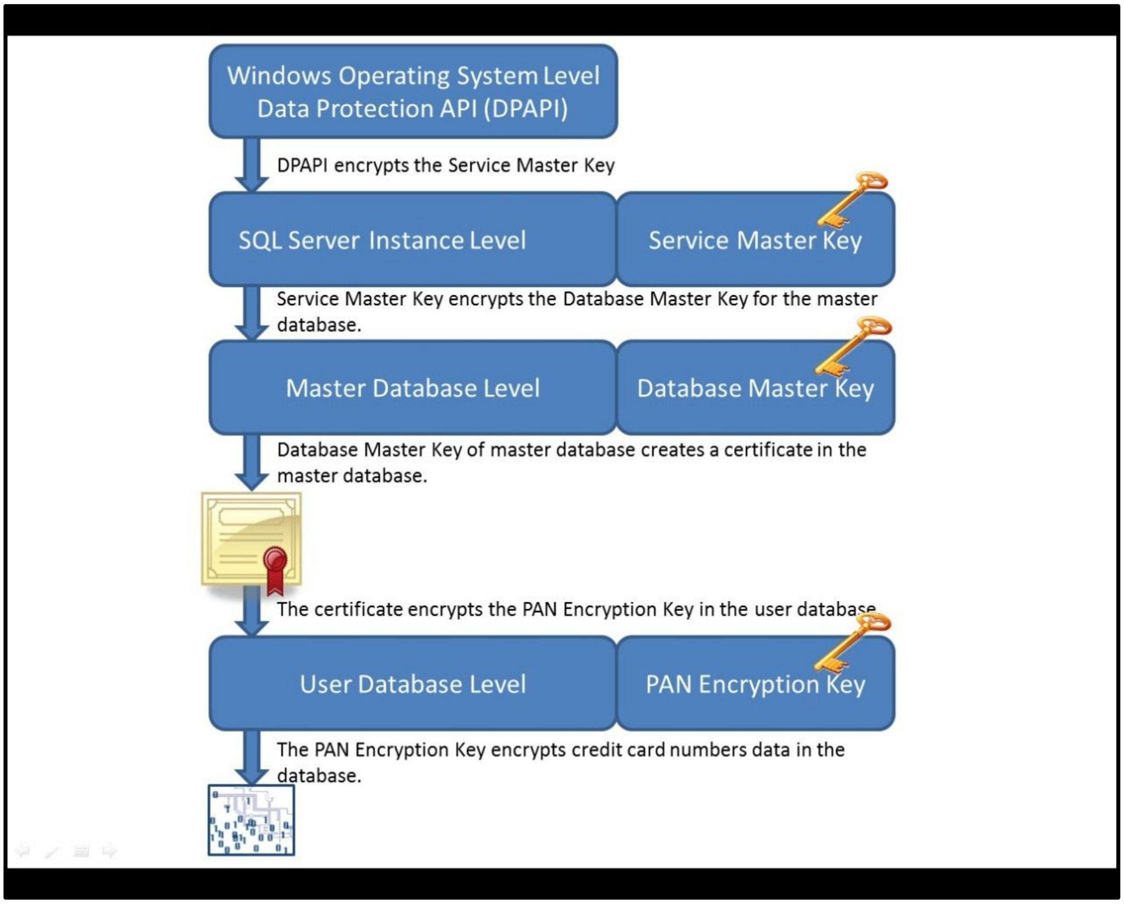
SQL Server has two primary applications for keys: a service master key (SMK) generated on and for a SQL Server instance and a database master key (DMK) used for a database.

Service Master Key: The SMK is automatically generated the first time the SQL Server instance is started and is used to encrypt a linked server password, credentials, and the database master key. The SMK is encrypted by using the local computer key using the Windows Data Protection API (DPAPI). The DPAPI uses a key that is derived from the Windows credentials of the SQL Server service account and the computer's credentials. The SMK can only be decrypted by the service account under which it was created or by a principal that has access to the machine's credentials.

Database Master Key: The database master key is a symmetric key that is used to protect the private keys of certificates and asymmetric keys that are present in the database. It can also be used to encrypt data, but it has length limitations that make it less practical for data than using another symmetric key. When the DMK is created, it is encrypted by using the Triple DES algorithm (For SQL 2008 and SQL 2008 R2) or using AES algorithm (For SQL 2012) and a user-supplied password. To enable the automatic decryption of the DMK, a copy of the key is encrypted by using the SMK. It is stored in both the database where it is used and in the master system database. The copy of the DMK stored in the master system database is silently updated whenever the DMK is changed. However, this default can be changed by using the DROP ENCRYPTION BY SERVICE MASTER KEY option of the ALTER MASTER KEY statement. A DMK that is not encrypted by the service master key must be opened by using the OPEN MASTER KEY statement and a password.

Accessing objects secured by the service master key requires either the SQL Server Service account that was used to create the key or the computer (machine) account. That is, the computer is tied to the system where the key was created. You can change the SQL Server Service account or the computer account without losing access to the key. However, if you change both, you will lose access to the service master key. If you lose access to the service master key without one of these two elements, you be unable to decrypt data and objects encrypted by using the original key. Connections secured with the service master key cannot be restored without the service master key. Access to objects and data secured with the database master key require only the password that is used to help secure the key.

By making use of SQL Server's public key certificate, Aptify has provided a mechanism for encrypting the PAN encryption security key with a segregated encryption key. Aptify makes use of the SQL Server's certificates which is a digitally signed statement that holds the encryption key and links this to the identity of a user. This certificate is created with the use of the DMK, and in turn is used to encrypt the PAN encryption key.



Windows Operating System Level Data Protection API (DPAPI)
DPAPI encrypts the Service Master Key
SQL Server Instance Level
Service Master Key
Service Master Key encrypts the Database Master Key for the master database.
Master Database Level
Database Master Key
Database Master Key of master database creates a certificate in the master database.
The certificate encrypts the PAN Encryption Key in the user database
User Database Level
PAN Encryption Key
The PAN Encryption Key encrypts credit card numbers data in the database.

Figure 4: Encryption Keys Architecture

To set up a SQL encrypted encryption key, the following instructions should be followed:

1. A Service Master Key is generated automatically the first time it is needed to encrypt another key. In case the Service Master Key is not automatically generated based on the SQL server version, use the create SQL statement to create it once. Altering or restoring the Service Master Key involves decrypting and re-encrypting the complete encryption hierarchy. Unless the key has been compromised, this resource-intensive operation should be avoided or scheduled during a period of low demand.
   - As a System Administrative user of the Aptify database, run the following SQL statement, replacing the password value with a password which follows secure password practices.

     ```
     -- Create a Service Master Key
     USE MASTER;
     GO
     CREATE MASTER KEY ENCRYPTION BY
     PASSWORD = 'YourOrganizationsSecurePassword'
     GO
     ```

   - As a System Administrative user of the Master database, run the following SQL statement if you want to regenerate Service Master Key.

     ```
     -- Regenerate Service Master Key
     ALTER SERVICE MASTER KEY REGENERATE

     GO
     ```

     NOTE: Do not use this statement unless it is absolutely necessary. This statement will generate a random new SMK, will decrypt the data encrypted using the current SMK, and will re-encrypt it using the newly generated SMK. This statement will either succeed in generating a new SMK or it will fail with an error without changing any data.

2. A Database Master Key (DMK) must be created upon the Aptify database. This only needs to be done once.

   - As a System Administrative user of the Aptify database, run the following SQL statement, replacing the password value with a password which follows secure password practices.

     ```
     -- Create a Database Master Key
     USE Aptify;
     GO
     CREATE MASTER KEY ENCRYPTION BY
     PASSWORD = 'YourOrganizationsSecurePassword'
     ```

GO

IMPORTANT! This password will need to be saved in a secure location for use in the event of a system restore. If this password is lost, it will be impossible to decrypt the encryption key that protects the PANs, and the PANs will be unrecoverable. Sees the "Database Master Key Backup and Restoration Procedures***" below for instruction on how to securely backup and restore the key.***

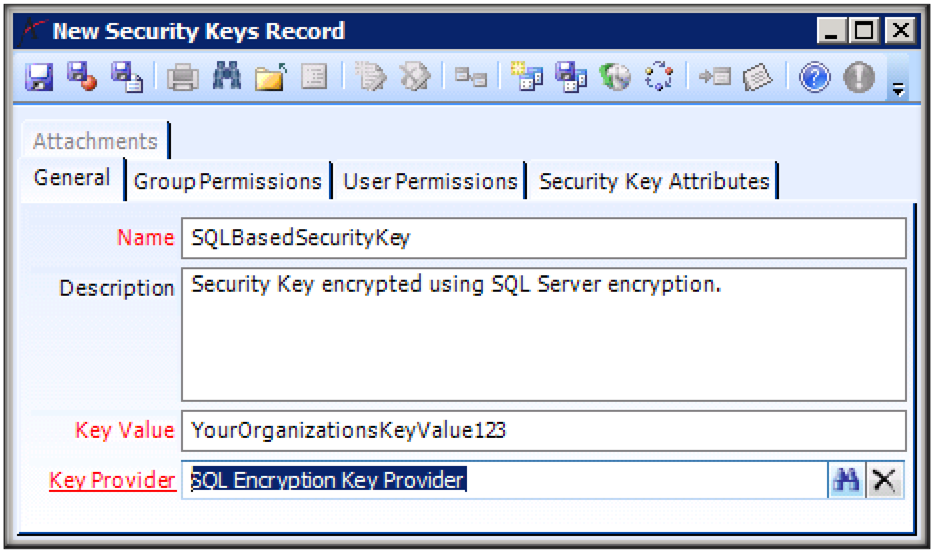
3. As an administrative Aptify user with system administrative rights, open a new Security Keys record within Aptify. This is the PAN security key that is used to encrypt the credit card numbers.

    - This is found in the Security Keys service within the Framework application by default.
    - A user identified as a System Administrator in the Aptify User Administration Wizard is a DBOwner in SQL Server and not a Sys Admin. The account that will create security keys must be manually identified as a Sys Admin in the *Security > Logins* area of SQL Server Management Studio.

4. Provide a **Name** and **Key Value**.

    - The Key Value should be treated as a password when determining the value.

5. Remove the current value within the **Key Provider** field, and enter **SQL Encryption Key Provider**.



Name: SQLBasedSecurityKey

Description: Security Key encrypted using SQL Server encryption.

Key Value: YourOrganizationsKeyValue123

Key Provider: SQL Encryption Key Provider

Figure 5 - Security Keys Record

6. Configure the User and/or Group permissions.

   - Click the **Group Permissions** tab and add Groups, as necessary

   - Click the **User Permissions** tab and add Users, as necessary.

   - Be sure to only include users and groups which are directly involved in taking and processing Orders from customers and those users or groups with an accounting function that require access to payments.

7. Select the Security Key Attributes tab. Provide values for the **EncryptProvidedCertName** and **CertificateSubject** attributes.

   - **EncryptProvidedCertName:** Enter a name for the certificate that the system will generate when encrypting the Key Value, which will be used to encrypt the credit cards.

   - **CertificateSubject:** Enter a description of the certificate. This is used to identify the certificate's usage.

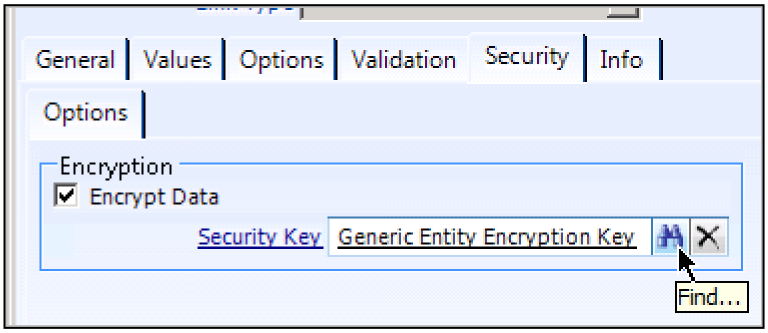
8. Save and close this record.

9. Open the Payment Information Entities record to replace the default security key with the newly created one.

   - The Entities service is found within the Framework application.

10. On the Fields tab, select the **CCAccountNumber** field and open it.

11. On the Security's tab of this field, select the Find dialog to update the Security Key value with the name of the newly created key.



Encryption

☑ Encrypt Data

Security Key: Generic Entity Encryption Key

Figure 6 - CCAccountNumber Field's Security Tab

12. Select **OK** to close this field once the Security Key value has been updated.

13. To update all existing records to use this new key, select the **Save** button in the data control bar of the Payment Information Entities record.

The save process may take a considerable amount of time, depending on the number of Payment Information records existing in the database and it is recommended that you run this process during a low peak time on your system.

This process, excluding the first and second steps, should be followed every time the Security Key is rotated (see "Key Rotation" section below). It is also recommended that this procedure should be done after credit cards have been purged to limit the number of records which must be updated (see "Using the Credit Card Purge Utility" section).

3.6.2.1 Database Master Key Backup and Restoration Procedures

It is important to create a backup of the key, and store it in a secure location off site. To do this, you will create an encrypted file and save it to a location accessible directly by the SQL Server. The password used to encrypt the key should be different than the one used to create the DMK.

To create a backup use the following SQL:

```
BACKUP MASTER KEY TO FILE = 'D:\SomeDirectory\MasterKeyFileName'
ENCRYPTION BY PASSWORD = 'ADifferentPassword';
GO
```

If it is necessary to restore the Aptify Database to a new SQL Server, the Database Master Key must be restored prior to restoring the Aptify Database. The decryption password is the password used to encrypt the file, and the encryption password is the original password used when creating the key, and to encrypt it on the database.

To restore the Database Master Key, use the following SQL:

```
RESTORE DATABASE MASTER KEY

    FROM FILE = 'e:\SomeDirectory\MasterKeyFileName'
    DECRYPTION BY PASSWORD = 'ADifferentPassword'
    ENCRYPTION BY PASSWORD = 'APassword';
    GO
```

For more information about creating, backing up and restoring Master Keys, see the following Microsoft references:

- Creating Master Keys: http://technet.microsoft.com/en-us/library/ms174382.aspx
- Backing up Master Keys: http://technet.microsoft.com/en-us/library/ms174387.aspx
- Restoring Master Keys: http://technet.microsoft.com/en-us/library/ms186336.aspx

3.6.3 Key Rotation

It is highly recommended that the PAN encryption key must be changed regularly. Aptify recommends changing PAN Encryption key at least annually and additionally it should be retired or replaced at any time when the integrity of the key has been weakened, or there is a known or suspected compromise of a key. Please follow the instructions given in "Implementing Key Encryption using SQL Server" section for rotating the keys. As noted previously, this task can follow immediately after purging cards that have exceeded the retention period. Changing the security key can be a time consuming process and it is highly recommended that a database backup be created first. See the *Aptify 5.0 Administrators Guide* for more information about database backup procedures. Aptify recommends keeping a history of all the cryptographic keys and making sure that none of the replaced or retired keys are used again. Also, make sure that Database Master Key and Service Master Key are also changed whenever there is a suspected security compromise, or the Key custodian leaves the organization. Please use the Sample Key Custodian Form to manage Key Custodians.

3.6.4 Secure Wipe of Old Cryptographic Data

Credit Card Purge Utility must be used to securely wipe the credit card data (Instructions for purging cards are available in section "Using the Credit Card Purge Utility"). For SMK and DMK, replacing of the Key will overwrite the older Key, thus making the older key non retrievable. Old PAN Encryption key should be deleted once it has been rotated by a new PAN Encryption Key. Deletion of PAN Encryption Key can be done by deleting the Security Keys record for the particular PAN Encryption Key. Only a person who has the rights to rotate keys should be given access to delete the keys. System will not allow deletion of a Key if it is currently in use. After purging credit card data or after deleting Keys, it is possible that remnants of the old data still exist on the file system of the OS in locations discarded by SQL Server. Aptify has no control over how SQL Server or the file system manages storage, so it is recommended that a secure wipe tool or a data scrubber such as Eraser (http://eraser.heidi.ie/) be used in case the storage media is to be physically discarded. Instructions on how to use eraser are available on eraser's website (http://eraser.heidi.ie/documentation/). Industry accepted deletion standards such as DoD 5220.22-M should be adhered to for getting rid of Data Remanence.

3.7 Secure Authentication and Unique User IDs

PA-DSS requirement: 3.1, 3.2
PCI-DSS requirement: 2.1, 8.1, 8.2, 8.5

Aptify authentication is based on SQL Server user security; each Aptify user corresponds to a single SQL Server login. All users must have a unique user ID to log into Aptify, and sharing login information violates Aptify licensing policy and will compromise security and violate PCI guidelines. The recommended method of user validation is to make use of Microsoft Windows Integrated Security. This links the Aptify user to the organization's domain security and makes it easier to administer Aptify user accounts. The other option is to allow the creation of SQL logins which will control the access and permissions to tables within Aptify. Note that SQL Server 2012 and 2008 require strong passwords to be used when creating new users, and Aptify also requires that an administrator provide a strong password at the time the user is created and whenever the password is changed.

Aptify leverages Microsoft SQL Server and employs many of SQL Server's security features. However, SQL Server supports a system administrator (sa) user who has full permissions to the entire database. The sa account should not be used to modify or administer Aptify, and should be deactivated or renamed after the initial setup of the system. This account must be a restricted account with only a specific person having access to it. There are no default admin accounts that can be used with Aptify.

In order to maintain PCI DSS compliance, any changes made to the authentication configurations would need to be verified, so that it provides an authentication method that is at least as rigorous as PCI DSS requirements.

Access to any PC, server, and database running Aptify must require a unique user ID and be controlled by secure authentication and customers are strongly advised not to make use of shared logins to gain access to any resource running Aptify or any payment application. In addition, PCI standards demand that sufficient IT policies are in place to enforce strong passwords that are changed frequently.

Aptify also strongly recommends implementing password policies and procedures, including instituting a domain policy with appropriate password complexity rules, rotation schedules, and lockout/reset procedures in accordance with PCI-DSS requirement 8. PCI Guidelines for a secure password management are:

- Require the password to be a minimum of seven characters in length and to contain both numeric and alphabetic characters.
- Require the password to be changed at least every 90 days.
- Password history should be kept and a new password is required to be different than any of the last four passwords used.
- Repeated access attempts should be limited by locking out the user account after not more than six logon attempts.
- Lockout duration should be a minimum of 30 minutes or until an administrator enables the user ID.
- If session has been idle for more than 15 minutes, the application should require the user to re-authenticate to re-activate the session.

Additional information for implementing strong password policies can be found here:

http://technet.microsoft.com/en-us/library/cc875814.aspx

http://technet.microsoft.com/en-us/library/ms161959.aspx

3.8 Logging User Access

PA-DSS requirement: 4.1, 4.2
PCI-DSS requirement: 10.1, 10.2, 10.3

This set of requirements focus on ensuring that appropriate logging of user activity is captured in the payment application to support audit procedures to determine which users viewed or modified any part of the system.

These requirements are met in several ways. Aptify captures record history for any changes made to the system. This feature is enabled in Aptify by default and there is no configuration required to set it up. This record history includes changes made by users to data through the application as well as schema or metadata changes made through the Aptify platform made by developers. While administrators can selectively turn record history off, customers are strongly discouraged from disabling record history unless other mechanisms exist to capture this information in a PCI-compliant manner or risk becoming non-compliant with PCI-DSS.

Aptify record history does not capture some PCI-relevant information, including user logins and logouts, invalid access attempts, “read” access to sensitive data, or deletion/manipulation of log data that may exist outside of Aptify. In addition, users with SQL Server logins can bypass the Aptify client and access the database directly using standard SQL Server tools. For these types of activities, Aptify recommends installing 3rd party tools designed to monitor and log database and file system activity originating both inside and outside of the Aptify client software.

Suggested tools include:

- **Microsoft SQL Server Audit:** This tool is included with SQL Server 2008 Enterprise edition and is recommended by Microsoft and others as a capable tool for capturing PCI-relevant user activity. Specific rules can be created to focus on only those activities that are relevant to PCI requirements. Information about SQL Server Audit can be found here:

  http://msdn.microsoft.com/en-us/library/cc280386.aspx.

  An excellent whitepaper by Parentebeard LLC details how SQL Server can be configured to meet PCI auditing requirements. More information on this whitepaper can be found in the here.

The audit functionality in SQL Server allows for granular control over what is logged. With this feature, actions, tables and users are auditable. In the event of a system compromise, auditing is critical in researching activity associated with a particular scenario. The ability to implicitly log and capture user access to cardholder data, detect changes to database objects/stored procedures, identify changes to server configuration settings and detect modifications to audit configuration settings (e.g., changes to audits and audit specifications) is key to achieving PCI compliance.

When considering what to audit, it is important to balance the need to determine what was accessed and/ or manipulated against the amount of data being kept. On a database with high transaction volume, an audit log can rapidly grow to be many times the size of its source database. Keep in mind that an audit trail history must be retained for one year. The guidelines below include minimum data requirement for audit as per the PCI DSS.

First, Aptify recommends that the following system activities be audited or logged:

- **Login attempts** - both successful and failed login attempts
- **Server configuration** - changes to encryption keys, creation, deletion or modification of logins and server level permissions, creation and deletion of databases
- **Database** - creation, deletion or modification of schema objects such as tables, stored procedures and views, addition or deletion of roles and users and changes to their permissions
- **Data** - auditing of any actions, inserts, deletions, updates or selects against the Payment Information table, which contains cardholder data

Audits should be configured in such a way as to prevent tampering from SQL Server users, including members of the sysadmin fixed server role and from Windows users who may try to access the audit files without using SQL Server. To accomplish this, Aptify suggests placing audit files in folders or file shares that are not accessible to members of the SQL Server sysadmin role and end users.

Additionally, Aptify suggests giving the SQL Server service account only write access capability to the audit files folder and auditing actions against audits. Examples of actions to audit include changes to audit specifications and enabling or disabling of audits (SQL Server 2008 R2 and 2012 implicitly does this) and configuring audits to shut down the server if the audit fails. To do this, specify the ON_FAILURE = SHUTDOWN audit option when creating a server audit.

Another option is to write to the Windows Security event log. This allows for increased security over the log data but has a number of potential downsides:

- The security log contains all Windows security log events, not just those for SQL Server, thus making the detection of SQL Server issues more difficult
- Using the event log is slower than writing to a file and can affect server performance
- If the event log is set to overwrite when full, then a malicious user could purposefully fill the event log to cover their tracks

Consider the following to mitigate these risks:

- Send high volume audit information to a file and create another audit to send infrequent but important audit information, e.g., modifying the audit configuration, to the security log
- Frequently send the security log data to a separate secure facility
- Review Security log best practices documented by Microsoft at this TechNet article

It is recommended that, if using the Windows Security log to record SQL Server audit data, the Audit Collection Services of System Center Operations Manager be utilized to securely collect and store audit data outside of the log. Second, audit specifications should be defined on the server. These audit groups should be specified on SQL Server for any PCI compliant server as follows:

- SUCCESSFUL_LOGIN_GROUP
- LOGOUT_GROUP
- FAILED_LOGIN_GROUP
- LOGIN_CHANGE_PASSWORD_GROUP
- SERVER_ROLE_MEMBER_CHANGE_GROUP
- BACKUP_RESTORE_GROUP
- DBCC_GROUP
- SERVER_OPERATION_GROUP
- AUDIT_CHANGE_GROUP
- SERVER_STATE_CHANGE_GROUP
- SERVER_OBJECT_CHANGE_GROUP
- SERVER_PRINCIPAL_CHANGE_GROUP
- SERVER_PRINCIPAL_IMPERSONATION_GROUP
- SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
- SERVER_PERMISSION_CHANGE_GROUP
- SERVER_OBJECT_PERMISSION_CHANGE_ GROUP

Database audit specifications should be defined. These audit groups should be applied to any PCI compliant databases and should apply to all users:

- APPLICATION_ROLE_CHANGE_PASSWORD_ GROUP
- DATABASE_CHANGE_GROUP
- DATABASE_OWNERSHIP_CHANGE_GROUP

- SCHEMA_OBJECT_CHANGE_GROUP
- DATABASE_PERMISSION_CHANGE_GROUP
- DATABASE_OBJECT_ACCESS_GROUP

The audit groups listed below should be applied to any table in PCI compliant databases and should apply to all users:

- SELECT (Note that SELECT audits capture the SELECT statement and not the resulting data)
- INSERT
- UPDATE
- DELETE

Because the Database Engine can access the audit file, SQL Server logins with CONTROL SERVER permission can use the Database Engine to access the audit files. Define an audit on *master.sys.fn_get_audit_file* to record anyone reading the audit file. This will record which logins with CONTROL SERVER permission have accessed the audit file through SQL Server.

Although the audit specifications presented above will audit actions, such as inserts updates and deletes, actual changes to data are not audited. To do this, you should enable the Change Data Capture feature against any table containing cardholder data. Change Data Capture creates a change data capture table instance for each table being captured.

The Change Data Capture instance table contains all of the columns in the source table as well as five metadata columns to store information about the change. Unlike trigger based methods for capturing changes, Change Data Capture is implemented asynchronously using the log file and so have a far smaller effect on performance than trigger based methods.

Lastly, it is important to use Windows Authentication, which is Aptify's recommended approach, or at least an individual SQL Server login for each user as the audit functionality is dependent upon identifying the logged in user in the audit log. If a single application login or shared login is used then identifying a particular user making changes will be impossible. Using Windows authentication may make it easier to link and trace actions beyond the boundaries of the SQL Server.

3.8.1 Some Examples

Using SQL tools to monitor Database Access (All examples are for SQL Server 2012 using Transact SQL – though possible for other versions and using SSMS interface- and must be setup by a System Admin)

Payment application must provide automated audit trails to reconstruct the following events:

1. Verify all individual access to cardholder data through the payment application is logged.
    - All access to payment information must connect with the Payment Information Table
    - SQL Server offers Auditing for this. Audited events can be written to the event logs or to audit files.
    - Use the following SQL to audit any read access on Payment Information table:

```
USE master;
GO
CREATE SERVER AUDIT Aptify_PADSS_Audit
```

```
    TO FILE (FILEPATH = 'C:\Program Files\Microsoft SQL
Server\MSSQL11.MSSQLSERVER\MSSQL\DATA' );
GO
ALTER SERVER AUDIT Aptify_PADSS_Audit
    WITH (STATE = ON);
GO

USE Aptify;
GO
CREATE DATABASE AUDIT SPECIFICATION Audit_PaymentInfo_Tables
    FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (SELECT ON PaymentInformation BY public)
    WITH (STATE = ON);
GO
```

2. Verify actions taken by any individual with administrative privileges to the payment application are logged.
    - A specification can be added to the above Database Audit to track any changes to the payment Information Table by an Administrator.
    - Example:

```
USE Aptify;
GO
CREATE DATABASE AUDIT SPECIFICATION AuditAdmin_PaymentInfo_Tables
    FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (SELECT, UPDATE, INSERT ON PaymentInformation BY sysadmin)
    WITH (STATE = ON);
GO
```

3. Verify access to application audit trails managed by or within the application is logged.
    - The Aptify Application does not have the ability to manage or Audit these files.
4. Verify invalid logical access attempts are logged.
    - A specification can be created for the above Server Audit to track failed logins at both the server and database level
    - Example:

```
USE master;
CREATE SERVER AUDIT SPECIFICATION Login_DBAudit_Specification
FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (FAILED_LOGIN_GROUP);
GO

USE Aptify;
GO
CREATE DATABASE AUDIT SPECIFICATION Login_AptifyAudit_Specification
    FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (FAILED_DATABASE_AUTHENTICATION_GROUP)
    WITH (STATE = ON);
```

```
GO
```

5. Verify use of and changes to the payment application's identification and authentication mechanisms (including but not limited to creation of new accounts, elevation of privileges, etc.), and all changes, additions, deletions to application accounts with root or administrative privileges are logged.
    - An Audit specification can be added to the above Server Audit which can track the permission changes of database objects, groups, and roles as well as memberships within Groups, Roles, and Principles:

```
USE master;
CREATE SERVER AUDIT SPECIFICATION Permission_Audit_Specification
FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (DATABASE_OBJECT_PERMISSION_CHANGE_GROUP),
    ADD (DATABASE_PERMISSION_CHANGE_GROUP),
    ADD (DATABASE_PRINCIPAL_CHANGE_GROUP),
    ADD (DATABASE_ROLE_MEMBER_CHANGE_GROUP),
    ADD (SERVER_ROLE_MEMBER_CHANGE_GROUP);
GO
```

6. Verify the following are logged:
    - Initialization of application audit logs
    - Stopping or pausing of application audit logs.
    - A specification can be added to the above Server Audit to track changes in Audits
    - Example:

```
CREATE SERVER AUDIT SPECIFICATION AuditChange_Audit_Specification
FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (AUDIT_CHANGE_GROUP);
GO
```

7. Verify the creation and deletion of system-level objects within or by the application is logged.
    - A Database level Audit can be added to the above Server Audit to track changes in all objects contained within the DB, including Creation and Deletion of these objects.
    - Example:

```
USE Aptify;
GO
CREATE DATABASE AUDIT SPECIFICATION Obj_CreateDelete_Specification
    FOR SERVER AUDIT Aptify_PADSS_Audit
    ADD (DATABASE_OBJECT_CHANGE_GROUP)
    WITH (STATE = ON);
GO
```

A SQL Server Audit log includes the Identity of the user triggering the event, the type of event, the date and time of the trigger, and the origin of the event. This will also detail what object(s) and data were affected. The source of the connection will also be recorded. The record of the event will also include whether a change was successful or not, except with the failed Login which will always be a failure.

- **Other 3rd Party SQL Logging Tools:** There are many tools available costing from $100's to $1,000's that may be suitable, however, Aptify has not tested or recommended any particular tools.

- **TripWire:** Tripwire is recognized as an industry leader in enterprise file integrity and compliance policy management. It is capable of monitoring file access and activity outside of SQL Server. This solution and others like it are commonly used to satisfy PCI audit logging requirements. See www.tripwire.com for more information.

3.9 Versioning Methodology

PA-DSS requirement: 5.4.4
PCI-DSS requirement: 6.7

Aptify application versioning methodology follows the procedures mentioned in PA-DSS Program guide. Application Release Notes contain the changes incorporated in the specific version of application release, security impact and how the changes will affect the product version. The Release Notes are published along with the application release and are communicated to the customers.

Application versioning methodology is explained in this PA-DSS implementation guide, and customers should read and understand the versioning methodology which will help them to identify if the payment application version in use is PA-DSS Validated Version.

Customer can check the PA-DSS Application validation status in the PCI Council Website, under the Section List of Validated Payment Application to ensure the version of application in use is PA-DSS Certified.

Aptify has a clearly defined product versioning methodology which defines a Major Version, a Minor Version, and a Service Pack Identifier. Only a manager can approve the Issue to be included in the build and they are required to make sure that each build adheres to the versioning methodology. We have a three digit version number (x.y.z) where the three numbers are separated by a dot. Aptify does not use the Wildcard elements. Product Versions are defined as following:

- **Major Version:** This includes major architecture changes and security-impacting changes in the platform. The first digit of the product version number defines the Major Version. For example, in 5.x.y, 5 is the Major Version.
- **Minor Version:** This includes the implementation of a large number of new features or one particularly complex feature, but has the same architectural model as the corresponding major version. Minor version can also include security-impacting changes. The second digit of the

product version number identifies the Minor Version. For example, in x.5.y, 5 is the Minor Version.

- **Service Pack Identifier:** This includes bug fixes and minimal new functionality, and does not include any security-impacting changes. The third digit of the product version is the Service Pack Identifiers. For example, in x.y.1, 1 is the Service Pack Identifier.

3.10 Use of Wireless Technology

PA-DSS requirement: 6.1, 6.2, 6.3
PCI-DSS requirement: 1.2.3, 2.1.1, 4.1.1

Aptify does not bundle or utilize any wireless technologies. However, if an organization has established a wireless network in its facilities, then that network should have the proper security safeguards enabled, such as firewalls, data encryption, and appropriate wireless authentication and encryption to prevent un-authorized access.

- **Use a network security key and change the default encryption keys**

If you have a wireless network, you should set up a network security key, which turns on encryption. With encryption, people can't connect to your network without the security key. Also, any information that's sent across your network is encrypted so that only computers that have the key to decrypt the information can read it. This can help avert attempts to access your network and files without your permission. Wi-Fi Protected Access (WPA or WPA2) is the recommended wireless network encryption method.

NOTE: We recommend using WPA2, if possible. We don't recommend using WEP for network security. WPA or WPA2 are more secure. If you try WPA or WPA2 and they don't work, we recommend that you upgrade your network adapter to one that works with WPA or WPA2.

- **Change the default administrator name and password on your router or access point**

If you have a router or access point, you probably used a default name and password to set up the equipment. Most manufacturers use the same default name and password for all of their equipment, which someone could use to access your router or access point without your knowledge. To avoid that risk, change the default administrator user name and password for your router. Check the information that came with your device for instructions about how to change the name and password.

- **Change the default SSID and SNMP Community Strings**

Routers and access points use a wireless network name known as a service set identifier (SSID). Most manufacturers use the same SSID for all of their routers and access points. We recommend that you change the default SSID to keep your wireless network from overlapping with other wireless networks that might be using the default SSID. It makes it easier for you to identify which wireless network is yours, if there's more than one nearby, because the SSID is typically shown in the list of available networks. Check the information that came with your device for instructions about how to change the default SSID.

- **Change wireless encryption keys, passwords and SNMP strings anytime anyone with knowledge of the keys/passwords leaves the company or changes positions**

All the wireless encryption keys, passwords and SNMP strings must be changed, anytime a person with knowledge of keys/passwords leaves the company or changes his/her position.

- **Install a firewall between any wireless networks and systems that store cardholder data**

A firewall must be installed between any wireless network and system that stores cardholder data. Firewall must be configured to deny or, if such traffic is necessary for business purposes, permit only authorized traffic between the wireless environment and the cardholder data environment.

- **Position your router or access point carefully**

Wireless signals can transmit a few hundred feet, so the signal from your network could be broadcast outside of your home. You can help limit the area that your wireless signal reaches by positioning your router or access point close to the center of your home rather than near an outside wall or window.

3.11 Use of Necessary and Secure Services, Ports and Protocols

PA-DSS requirement: 8.2
PCI-DSS requirement: 2.2.2

This requirement states that the payment application must only use or require use of necessary and secure services, protocols, daemons, components, and dependent software and hardware, including those provided by third parties, for any functionality of the payment application. For Aptify e-Business site ports 80 and 443 need to be open, per internet standards. PayPal Payflow also uses port 443 for secure transfer of sensitive information. It is also required to open port 1433, which is the default port for Microsoft SQL Server (It is possible for an organization to specify a different port for Microsoft SQL Server). All the necessary services required to run SQL server should also be running including SQL Server service and SQL Server Agent service. Please refer to the "Application Requirements" section for details on the required hardware.

3.12 Separate Web Server from Database Server

PA-DSS requirement: 9.1
PCI-DSS requirement: 1.3.7

The Aptify database server should never be deployed on the same server used to host your organization's web site nor should it be directly connected to the Internet. Connection to the Aptify database and the SQL Server which hosts it should be protected with the use of a firewall. In order to be PCI compliant, customers must use HTTP over a Secure Socket Layer (SSL) to secure the e-Business site's content that cannot be accessed without logging in first. If the customer runs a web server for e-Business, it should be installed in a DMZ and configured according to the installation instructions for e-Business given in the e-Business Install Guide.

3.13 Remote Software Updates

PA-DSS requirement: 10.2.1
PCI-DSS requirement: 12.3.9

PA-DSS and PCI-DSS requirements place controls on how software updates are delivered to customers. Aptify does not provide a "live" update service where software updates are delivered directly to the installed system.

Aptify distributes software releases and updates via a Secure FTP (SFTP) server with user logins restricted to customer-specific folders.

> It is highly recommended that customers download software updates to a computer on a network separate from the Aptify database. Once the update has been verified on a test system, the changes can then be migrated to the production system.

3.14 Remote Access to Aptify

PA-DSS requirement: 10.1, 10.2,
PCI-DSS requirement: 2, 4.1, 8, 10

If an organization intends to allow remote access to Aptify, it is important to configure remote support with adequate security measures, such as requiring unique user name and password information for each user, filtering access based on MAC or IP address, and requiring a Virtual Private Network (VPN) connection in order to access the Aptify database and other network resources from outside of the organization's infrastructure. All remote connectivity software and IT policies governing software that provides access to the database server must address the following requirements:

- encrypt transmission data
- lockout users after a certain number of failed login attempts
- enforce password complexity policies as specified by PCI standards
- enable logging (success and failed)

- prohibit shared user IDs
- restricting access to customer passwords to authorized personnel
- disable all default user IDs
- authenticate users through Windows Local or Active Directory account settings

PCI guidelines also state that 2-factor authentication must be employed if users connect to Aptify remotely. 2-factor authentication includes an authentication method such as smart card, token, or PIN in addition to a normal user ID and password. Aptify does not supply nor require 2-factor authentication, and does not interfere with any methods an IT organization may want to implement.

3.15 Encrypt Sensitive Traffic over Public Networks

PA-DSS requirement: 11.1
PCI-DSS requirement: 4.1

This requirement states that the application should support the use of strong cryptography and security protocols (for example SSL/TLSIOSEC, SSH, etc.) to safeguard sensitive cardholder date sent over public networks. In Aptify, secure data is transferred over public network while using e-Business and while connecting to PayPal Payflow (Payment Gateway). In e-Business, the one way data transfer happens over the internet when a customer makes a payment using the e-Business site. You must ensure usage of HTTP over a Secure Socket Layer (SSL), to secure the e-Business site's content that cannot be accessed without logging in first. This will create an HTTPS site for all sensitive areas of the website. Ensure that all cookies sent over HTTPS are also marked with the "secure" token, so that they are not sent in plain text at any point. Please refer to the e-Business Developer Guide for information on how to set up the site over HTTPS. Also modify the website to set the HTTPOnly attribute for all the cookies, to ensure that they are accessible over HTTP and not by scripts. Refer to your specific CMS's documentation for information on how to secure content with SSL.

Sensitive data is transferred over the internet to PayPal Payflow using their API over SSL via port 443. Please refer to PayPal Payflow Developers Guide and "Implementing Key Encryption using SQL Server section" for more information. If the customer/integrator decides to use other gateway than Paypal Payflow then they need to make sure that they use strong cryptography and security protocol when transmitting sensitive data. PCI requires strong password policy for end users to the e-Business website. Please refer to the Aptify e-Business Developer Guide for instructions on setting strong password policy. Aptify recommends to have auto complete tag and set it to "off" on all sensitive input fields in the website.

Clickjacking, also known as a "UI redress attack", is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to other another page, most likely owned by another application, domain, or both. Using a similar technique, keystrokes can also be hijacked. With a carefully crafted combination of stylesheets, iframes, and text boxes, a user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker. Aptify recommends taking counter measures to defend your website against Clickjacking. Please see the following whitepaper, jointly published by Stanford and Carnegie Mellon Universities in 2010, for an in depth analysis of click jacking, potential counter measures and their relative effectiveness (http://seclab.stanford.edu/websec/framebusting/framebust.pdf). For an alternate viewpoint on the issue and effective remedies, please see the OWASP article on click jacking and UI redressing (http://www.owasp.org/index.php/Clickjacking).

Sanitizing is the process which removes specific unrequired or undesired characters from a value by the user or application. By properly sanitizing input several of the OWASP top 10 concerns, including SQL injection and Cross Site Scripting, can be mitigated. Aptify strongly recommends to sanitize all data, for the web application, from untrusted sources such as a browser.

Also, Aptify strongly recommends that you turn on Custom Errors in your web.config before going live with e-Business site. By default, Sitefinity creates a web site with Custom Errors set to Off, which is useful for development purposes but can expose sensitive web site information when left Off on a public server. Refer to ASP.NET documentation for more information on the Customer Errors web.config property. Aptify also recommends disabling the 'Forgot UserID' functionality in order be to PCI compliant. Please contact Aptify support for assistance in disabling this e-Business functionality and to discuss possible alternate approaches.

3.16 Encrypt non-console Administrative Access

PA-DSS requirement: 12.1, 12.2
PCI-DSS requirement: 2.3

This requirement states that all non-console administrative access must be encrypted using technologies such as SSH, VPN, or SSL/TLS. Administrative functionality in Aptify is accomplished using the Smart Client application. If accessing Aptify remotely, you must use one of the encryption techniques listed above. By using any of the above mentioned techniques, like SSL wrapper, available in MS SQL servers, the application would provide an additional layer of protection against network eavesdropping.

> To ensure secure administrative access within the firewall, it is recommended that all SQL Server connections between the Smart Client and the database be encrypted using SSL. Instructions for configuring SSL database connections are available in the SQL Server documentation. No special Aptify settings or configurations are necessary to support this.

4 Other Best Practices

The previous chapter outlined the implementation and operational requirements for running Aptify in a PCI-compliant manner. In addition, there are other best practices Aptify specifically recommends.

4.1 Use of Cardholder Data

During the testing and development phases of operation, it is important that you refrain from using payment information from actual customers. If this information is required to solve a specific problem, then the amount of information used must be limited to the smallest subset that could be used to solve the issue, and the information should be stored within a secure location which will only allow access to those who need it to perform these tests. All sensitive information must be stored encrypted as it would be on the original system. After the completion of the testing, all information related to the customer must be deleted in accordance with best practices for securely removing data from the device on which the information was stored.

4.2 Aptify Access to Cardholder Data

There are a number of cases where Aptify will assist customers on a post go-live basis. To ensure adequate protection of cardholder data, there are specific recommendations that should be followed.

- During the support of your environment, a consultant assigned to your organization should only have access to customer's sensitive data on a limited scope, and only if required to complete the observation. All development should be performed within the development environment, which should not have any sensitive data included. Sensitive information would only be made available to a consultant for troubleshooting a problem which requires this access, and it should be supervised and limited to the time required to identify the problem.

- While making use of Aptify Support, it may be necessary to provide information which would allow the Support team to duplicate a reported problem you may be experiencing. If it is possible to provide testing data, this would be preferable, but if not, any information provided from the live system should be scrubbed of any sensitive information. Any information that is required to help resolve the issue must be limited to the smallest subset possible, and this information must be securely transmitted to Aptify Support. Aptify Support has been instructed in the proper use and storage of such information; it will be stored within a secure location with limited access and retained only long enough to identify the problem. Aptify's support team is strictly prohibited from collecting any Sensitive Authentication Data during troubleshooting.

4.3 User Access to Payment Information

Access to the Aptify payment processing functionality should only be granted to those individuals directly involved in taking and processing Orders from customers and to those users in an accounting function that require access to payments. It is important to limit access to these areas of the system to limit the number of people with potential access to the customer's sensitive data. This information is further protected by allowing only a subset of the users with permissions to the Payments module access to the Security Key, as described earlier. Limiting access is done through the use of permissions linked to each entity. By controlling the users and group permissions on the Payments and Orders Entity, you can limit and control access to the processes. See the “Entities Level Security” chapter of the *Aptify 5.0 Administration Guide* for more information.

4.4 Electronic Payments

Aptify recommends the use of PayPal PayFlow Pro as an Electronic Payment Processing service and a plug-in to the PayPal API is included within the default system. If using a payment processing service other than PayPal, it is important to ensure that this service meets the PCI requirements for encrypted transmission of data and secure processing of customer data. When integrating with a third-party payment processing service, follow the best practice recommendations of that service to properly secure sensitive data.

Appendix A

Legal Terms and Conditions

Per PCI regulation, Aptify is required to notify all purchasers, users, and other licensees of the following:

Acceptance of a given payment application by the PCI Security Standards Council, LLC (PCI SSC) only applies to the specific version of that payment application that was reviewed by a PA-QSA and subsequently accepted by PCI SSC (the "Accepted Version"). If any aspect of a payment application or version thereof is different from that which was reviewed by the PA-QSA and accepted by PCI SSC even if the different payment application or version (the "Alternate Version") conforms to the basic product description of the Accepted Version – then the Alternate Version should not be considered accepted by PCI SSC, nor promoted as accepted by PCI SSC.

No vendor or other third party may refer to a payment application as "PCI Approved" or "PCI SSC Approved", and no vendor or other third party may otherwise state or imply that PCI SSC has, in whole or part, accepted or approved any aspect of a vendor or its services or payment applications, except to the extent and subject to the terms and restrictions expressly set forth in a written agreement with PCI SSC, or in a PA-DSS letter of acceptance provided by PCI SSC. All other references to PCI SSC's approval or acceptance of a payment application or version thereof are strictly and actively prohibited by PCI SSC.

When granted, PCI SSC acceptance is provided to ensure certain security and operational characteristics important to the achievement of PCI SSC's goals, but such acceptance does not under any circumstances include or imply any endorsement or warranty regarding the payment application vendor or the functionality, quality, or performance of the payment application or any other product or service. PCI SSC does not warrant any products or services provided by third parties. PCI SSC acceptance does not, under any circumstances, include or imply any product warranties from PCI SSC, including, without limitation, any implied warranties of merchantability, fitness for purpose or noninfringement, all of which are expressly disclaimed by PCI SSC. All rights and remedies regarding products and services that have received acceptance from PCI SSC, shall be provided by the party providing such products or services, and not by PCI SSC or any payment brands.

Appendix B

Sample Key Custodian Form

The below is a sample of a form your key custodians should sign, customized for your company. All staff with authorization to access the encryption keys should be required to sign this.

As a condition of continued employment with ______________________________ and as an employee with responsibilities regarding the cryptographic security of confidential data, you are required to sign the following document to acknowledge those responsibilities.

The signer of this document is an employee with _____________________________ on the date shown below, with access to key management devices, software, and/or equipment, and hereby agrees that he/she:

- Has read and understood the policies and procedures associated with key management and agrees to comply with them to the best of their ability. He/she has had the opportunity to ask questions regarding this policy, and has had those questions answered to their satisfaction.
- Understands that non-compliance with these policies can lead to termination of employment and possible prosecution.
- Agrees to never divulge to any unauthorized party the key management practices or any related security systems, passwords, processes, or other secrets associated with the company's systems.
- Agrees to promptly report to management any suspicious activity, including but not limited to system or key compromise or theft.

I agree to the above in full and understand my responsibilities as indicated above.

Signed: ____

Printed Name: ____

Date: ____

Witnessed: ____

Printed Name: ____