Connected Payments
OpenEPS 829

PA-DSS 3.1 Implementation Guide

Version 2.7
July 2016

Address correspondence to:
Manager, Connected Payments
NCR Corporation
85 Argonaut, Suite 150
Aliso Viejo, CA 92656
Internet Address:
http://www.info.ncr.com/Feedback

Revision History

Date	Changed By	Comment	Version
06/24/2015	Terry A. Stevenson	Original	1.0
07/24/2015	Terry A. Stevenson	Update document to PA DSS 3.1	2.0
08/14/2015	Terry A. Stevenson	Updates	2.0
08/19/2015	Terry A. Stevenson	Updates recommended by Coalfire	2.1
08/20/2015	Terry A. Stevenson	Updates recommended by Coalfire	2.2
08/24/2015	Terry A. Stevenson	Updates recommended by Coalfire	2.3
08/26/2015	Terry A. Stevenson	Final updates recommended by Coalfire	2.4
12/23/2015	Terry A. Stevenson	Disabling CRL for P2PE only networks & Coalfire supporting letter	2.5
07/21/2016	Patrick Watson	Updated for 829.2.2*.2 Release Credit only support for Vx820	2.6
07/29/2016	Patrick Watson	Updated for 829.2.2*.5 Release Added full support for Vx820, Equinox L5200 & L5300	2.7

This implementation guide will be updated as necessary pursuant to PA DSS requirements and as NCR deems necessary.

The most recent copy of this document can be acquired by contacting NCR at ConnectedSupport@retalix.com. If you already have an account for the Connected Payments OpenEPS download site, you may download the latest version directly from there.

Notice

THE INFORMATION IN THIS DOCUMENT IS FOR INFORMATIONAL PURPOSES ONLY. NCR MAKES NO REPRESENTATION OR WARRANTY AS TO THE ACCURACY OR THE COMPLETENESS OF THE INFORMATION CONTAINED HEREIN. YOU ACKNOWLEDGE AND AGREE THAT THIS INFORMATION IS PROVIDED TO YOU ON THE CONDITION THAT NEITHER NCR NOR ANY OF ITS AFFILIATES OR REPRESENTATIVES WILL HAVE ANY LIABILITY IN RESPECT OF, OR AS A RESULT OF, THE USE OF THIS INFORMATION. IN ADDITION, YOU ACKNOWLEDGE AND AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR MAKING YOUR OWN DECISIONS BASED ON THE INFORMATION HEREIN.

Nothing herein shall be construed as limiting or reducing your obligations to comply with any applicable laws, regulations or industry standards relating to security or otherwise including, but not limited to PCI DSS and PA DSS.

The retailer may undertake activities that may affect compliance. For this reason, NCR is required to be specific to only the standard software provided by it.

Table of Contents

About this Document

This document describes the steps that must be followed in order for your OpenEPS client application installation[*s*] to comply with Payment Application – Data Security Standards (*PA-DSS*). The information in this document is based on PCI Security Standards Council Payment Application - Data Security Standards program (*version 3.1 dated April 2015*).

NCR instructs and advises its customers to deploy NCR applications in a manner that adheres to the PCI Data Security Standard (*v3.1*). Subsequent to this, best practices and hardening methods, such as those referenced by the Center for Internet Security (*CIS*), NIST/FIPS, and their various "Benchmarks", should be followed in order to enhance system logging, reduce the chance of intrusion and increase the ability to detect intrusion, as well as other general recommendations to secure networking environments. Such methods include, but are not limited to, enabling operating system auditing subsystems, system logging of individual servers to a centralized logging server, the disabling of infrequently-used or frequently vulnerable networking protocols and the implementation of certificate-based protocols for access to servers by users and vendors.

You must follow the steps outlined in this Implementation Guide in order for your OpenEPS client application installation to support your PCI DSS compliance efforts.

Executive Summary

OpenEPS, 829.*, has been Payment Application - Data Security Standard (*PA-DSS*) validated, in accordance with PA-DSS Version 3.1. For the PA-DSS assessment, we worked with the following PCI SSC approved Payment Application Qualified Security Assessor (*PAQSA*):

Coalfire Systems, Inc. 361 Centennial Parkway Suite 150 Louisville, CO 80027	Coalfire Systems, Inc. 1633 Westlake Ave N #100 Seattle, WA 98109

This document also explains the Payment Card Industry (*PCI*) initiative and the Payment Application Data Security Standard (*PA-DSS*) guidelines. The document then provides specific installation, configuration, and ongoing management best practices for using NCR's OpenEPS Version 829.* as a PA-DSS validated application operating in a PCI DSS compliant environment.

PCI Security Standards Council Reference Documents

The following documents provide additional detail surrounding the PCI SSC and related security programs (*PA-DSS, PCI DSS, et cetera*):

- Payment Card Industry Payment Applications - Data Security Standard (*PCI PA-DSS*)
  https://www.pcisecuritystandards.org/security_standards/index.php
- Payment Card Industry Data Security Standard (*PCI DSS*)
  https://www.pcisecuritystandards.org/security_standards/index.php
- Open Web Application Security Project (*OWASP*)
  http://www.owasp.org
- Center for Internet Security (*CIS*) Benchmarks (*used for OS Hardening*)
  https://benchmarks.cisecurity.org/downloads/multiform/
- National Institute of Standards and Technology (*used for cryptology & other security processes)*
  http://csrc.nist.gov/publications/PubsSPs.html
  http://csrc.nist.gov/publications/PubsFIPS.html

Application Summary

<table>
<tr><td>Payment Application Name</td><td>OpenEPS</td><td>Payment Application Version</td><td>829.*</td></tr>
<tr><td>Application Description</td><td colspan="3">OpenEPS is the client application for a payment service provider solution known as Connected Payments. OpenEPS interacts with the POS (point of sale) and clients' pinpads to provide the processing of encrypted credit card/debit (tender types) data. OpenEPS sends the encrypted data to NCR's Cloud solution (ServerEPS) for processing of the payment. There are other non-credit card/debit processing functions that OpenEPS provides for the POS and Connected Payments.</td></tr>
<tr><td>Typical Role of Application</td><td colspan="3">OpenEPS is a middleware application that interacts with the pinpad capturing the client's credit card/debit data and then sends that encrypted information to NCR's cloud for processing of the credit card/debit payment. OpenEPS resides on the client's POS located at the client's premise.</td></tr>
<tr><td>Target Market for Payment Application</td><td colspan="3">
<table>
<tr><td colspan="6">Target Market for Payment Application (check all that apply):</td></tr>
<tr><td>X</td><td>Retail</td><td>X</td><td>Processors</td><td></td><td>Gas/Oil</td></tr>
<tr><td></td><td>e-Commerce</td><td>X</td><td>Small/medium merchants</td><td></td><td></td></tr>
<tr><td>X</td><td colspan="5">Others (please specify): Brick and Mortar Merchants</td></tr>
</table>
</td></tr>
<tr><td>Stored Cardholder Data</td><td colspan="3">

The only location that credit card/debit data (prior to authorization) is stored in the OpenEPS application is in the offline file.

3DES DUKPT - DEK (3DES 168 bit encrypted message) message
OpenEPS – no communication with Connected Payments –
DEK Message is re-encrypted with AES 256 and stored in the offline file.
Store/forward – encrypted information is stored here until communications comes back and OpenEPS will continually try to send the messages up and overwrites the encrypted data.
OpenEPS <u>does not store any credit card/debit data.</u>

After the offline file has sent the encrypted transactional blob up to Connected Payments, the encrypted data is deleted by NULLS over the encrypted data and deleting the offline file. A new offline file is created after this process.
Offlines continually attempted to forward encrypted transactions to Connected Payments. If there are offlines that do not forward, a threshold is set (configurable) to send an alert when the threshold has been reach that the offline transactions are not being forwarded.
OpenEPS does not have access to any clear text PAN and the offline line file is encrypted with AES 256.

The entire transaction encrypted blob - Credit card information that is with encrypted 3DES DUKPT or AES 128 transactional data is then encrypted with AES 256 in the offline table (<u>DOUBLE ENCRYPTION</u>)
When the system is connected and on pre-authorization for offline or store and forward transactions are immediately sent to Connected Payments Cloud for processing.

</td></tr>
<tr><td>Components of the Payment Application</td><td colspan="3">The only component that is installed in a typical merchant environment is the OpenEPS DLL located on the POS. No modifications are allowed to change the OpenEPS DLL. There is integrity checking built into the application that is validated by Connected Payments Cloud to ensure the application has not been modified.</td></tr>
</table>

<table>
<tr><td>Required Third Party Payment Application Software</td><td>OpenEPS requires both a supported pinpad and a POS.

The following pinpads are supported by OpenEPS. All Pinpads must support Connected Payments Hardware Encryption (aka P2P).
<table>
<tr><th>Version</th><th>Supported Pinpads</th></tr>
<tr><td>829.2.2*.2</td><td>VeriFone Vx820 (credit only)</td></tr>
<tr><td>829.2.2*.5</td><td>VeriFone Vx820
Equinox L5200, L5300</td></tr>
</table></td></tr>
<tr><td>Database Software Supported</td><td>OpenEPS does not have a database. OpenEPS does not support traditional database software to store application state. It utilizes an internal proprietary flat file system.</td></tr>
<tr><td>Other Required Third Party Software</td><td>No 3rd party software applications are required by the payment application.</td></tr>
<tr><td>Operating System(s) Supported</td><td>Latest Supported Versions of:
POSReady 7
POSReady 2009 (with extended support)
Windows 7
Windows 8</td></tr>
<tr><td>Application Authentication</td><td>Authentication between the pinpad and OpenEPS
OpenEPS authentication between a pinpad and OpenEPS is through a session key.
Authentication between OpenEPS and the POS
None
Authentication between OpenEPS and ServerEPS
TLS 1.2 authentication with ServerEPS
Company number and store number validated by ServerEPS</td></tr>
<tr><td>Application Encryption</td><td><u>Hardware Encryption</u> – End to End
OpenEPS leverages a PTS approved hardware encryption solution using 3DES DUKPT
Offline files
OpenEPS further encrypts the entire 3DES DUKPT message with AES 256-bit encryption while stored in the offline file. Information is stored here until communication comes back and OpenEPS will continually try to send the messages up.
Manual entry would occur at the PINPAD.
<u>Enhance Software Encryption</u> -
OpenEPS exchanges keys with the pinpad so that the pinpad can encrypt the credit card data at the pinpad with AES.
Offline files
OpenEPS further encrypts the entire AES 128-bit message with AES 256-bit encryption while stored in the offline file. Information is stored here until communication comes back and OpenEPS will continually try to send the messages up (store/forward).
Manual entry would occur at the PINPAD and would be encrypted on the PINPAD.
If manual entry occurs on the POS, OpenEPS encrypts the data with the AES 128-bit.
AES message OpenEPS – no communication with Connected Payments – AES Message is re-encrypted with AES 256 and stored in the offline file. Information is stored here until</td></tr>
</table>

<table>
<tr><td></td><td>communications comes back and OpenEPS will continually try to send the messages up.</td></tr>
<tr><td>Application Functionality Supported</td><td>Payment Application Functionality (check only one):
<table>
<tr><td></td><td>Automated Fuel Dispenser</td><td></td><td>POS Kiosk</td><td></td><td>Payment Gateway/Switch</td></tr>
<tr><td></td><td>Card-Not-Present</td><td></td><td>POS Specialized</td><td>X</td><td>Payment Middleware</td></tr>
<tr><td></td><td>POS Admin</td><td></td><td>POS Suite/General</td><td></td><td>Payment Module</td></tr>
<tr><td></td><td>POS Face-to-Face/POI</td><td></td><td>Payment Back Office</td><td></td><td>Shopping Cart & Store Front</td></tr>
</table></td></tr>
<tr><td>Payment Processing Connections</td><td>Payment transactions are initiated by the cashier and/or customer at the point of sale. OpenEPS sends the information to the pinpad (amount, type of tender request et cetera) for a payment request. The customer conducts either a card swipe or the card is read to supply the credit/debit information. The pinpad encrypts the transactional data (credit/debit) using either P2P 3DES DUKPT or Enhance software AES 128 bit depending upon the pinpad utilized. Pinpad sends this encrypted information in the form of a blob to OpenEPS.

OpenEPS encapsulates the encrypted blob (transaction data) with message encryption (TLS 1.2, RSA/ECC 2048/224, SHA2, AES (GCM)) and send the information to Connected Payments for processing. Connected Payments, ServerEPS, decrypts the TLS message encryption, decrypts the 3DES DUKPT blob or AES blob, formats to the desire hosts requirements and sends out for authorization to the appropriate host.

Once Connected Payments receive the authorization message, then ServerEPS sends back to OpenEPS the response of approval/decline. OpenEPS then relays the information to the pinpad & POS.</td></tr>
<tr><td>Description of Listing Versioning Methodology</td><td>There are three main OpenEPS modules used. The OpenEPS modules are versioned based on a 4 quartet numbering scheme. x.x.x.x

• The first quartet designates the Primary Version of the code. This is updated when PCI impactful development changes are made.

• The second quartet designates the Secondary Version of the code. This is updated when non-PCI impactful development changes are made.

• The third quartet varies depending on the module. It is always 0 except for the MTX_EPS.dll. For that module it designates whether Hardware or Software encryption is being used and the primary and secondary data center usage. For clarification, this designation is only a flag for versioning and does not affect or change hardware or software encryption module; meaning this is only a label change and not a code change. This third quartet designates 8 different modules in order to support information being sent to specified different data centers.

• The forth quartet designates the specific build number.

<table>
<tr><td>Example</td><td>Primary Version</td><td>Secondary Version</td></tr>
</table></td></tr>
</table>

	MTX_EPS.dll	829.1.21.234	xxx	.	x	.	xx
	MTX_SE.dll	829.1.0.2	xxx	.	x	.	0
	MTX_POS.dll	829.1.0.1	xxx	.	x	.	0

Primary Version
- Updated for PCI impactful development changes

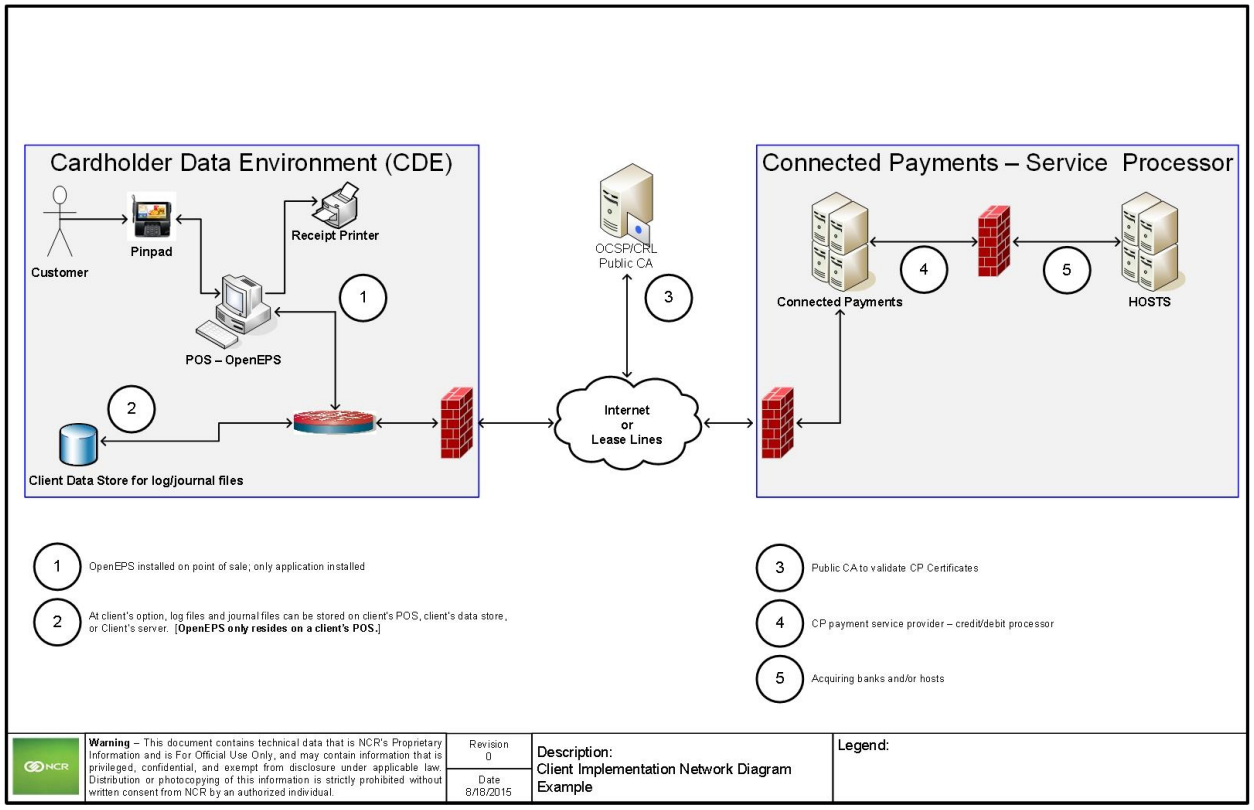
Secondary Version
- Updated for non PCI impactful development changes

Variable
- 0 for all but MTX_EPS.dll
- For MTX_EPS.dll
  - First Digit
    - 1=Software Encryption
    - 2 = Hardware P2P Encryption
  - Second Digit
    - 1=Primary DC=1, Secondary DC=2
    - 2=Primary DC=2, Secondary DC=1
    - 3=Primary DC=3, Secondary DC=4
    - 4=Primary DC=4, Secondary DC=3

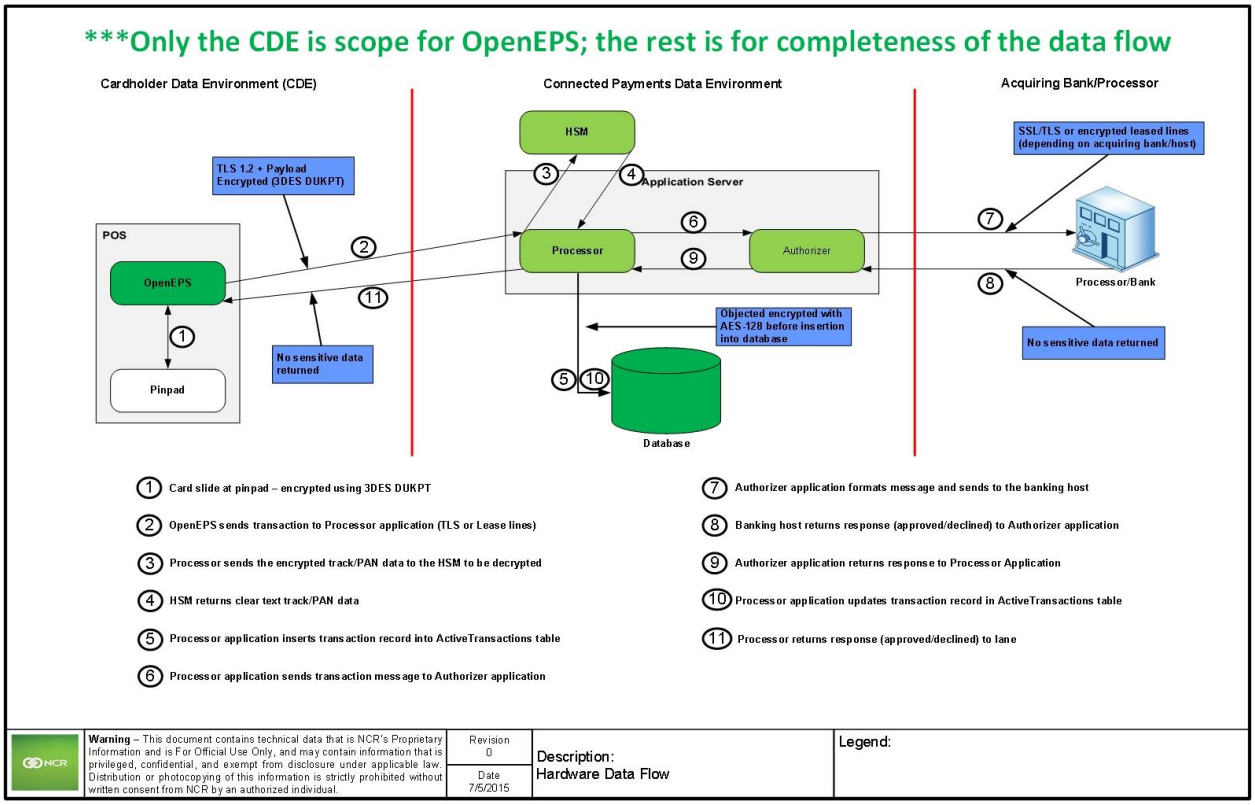
Build number
- Increased with each build.

Based on the above versioning methodology since all changes that would affect any PA-DSS control would trigger a change to the primary component of the version the listed version for PCI will use wildcarding and be Primary.* such as 829.*

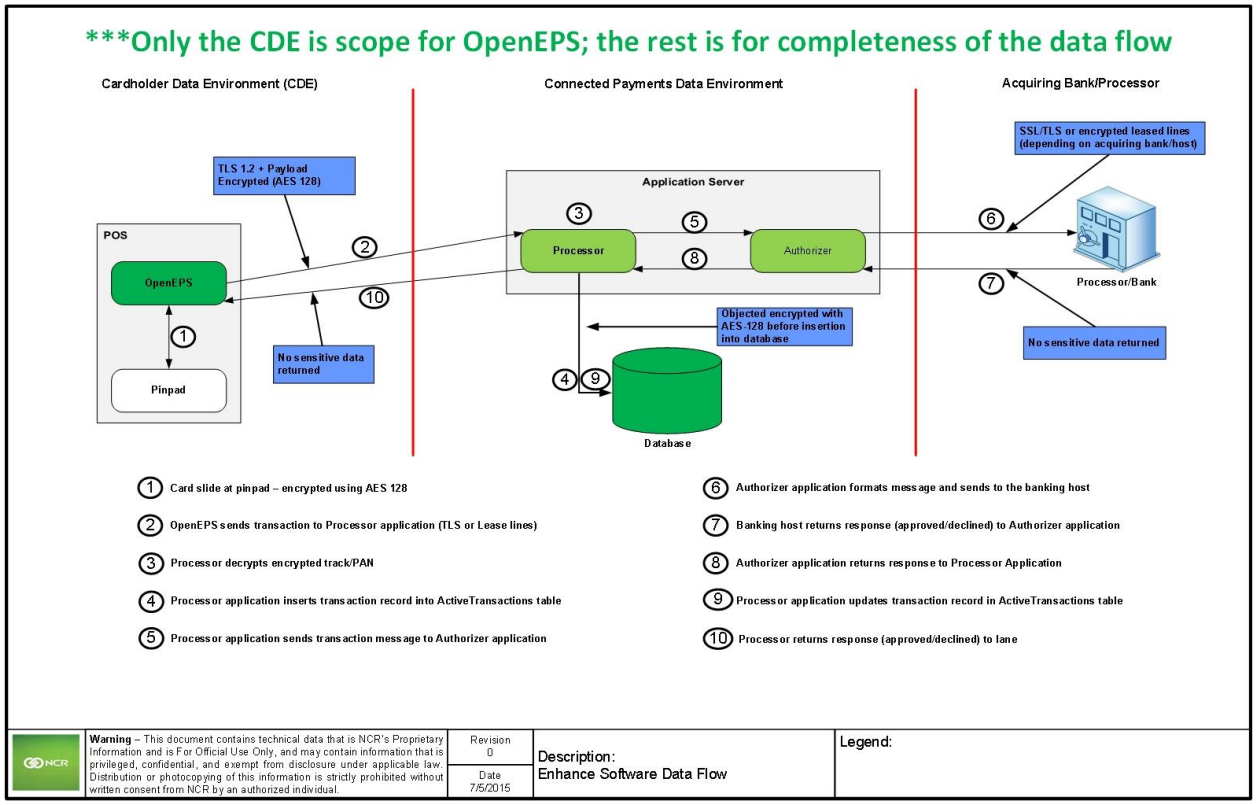
OpenEPS Network Diagram Example



OpenEPS Hardware Data Flow Diagram



OpenEPS Enhance Software Data Flow Diagram



Difference between PCI Compliance and PA-DSS Validation

As a software vendor who develops payment applications, our responsibility is to be "PA-DSS Validated." We have performed an assessment and payment application validation review with our independent assessment firm (*PAQSA*), to ensure that our platform does conform to industry best practices when handling, managing and storing payment related information. PA-DSS Version 3.1 is the standard against which Payment Application has been tested, assessed, and validated.

PCI Compliance is then later obtained by the merchant, and is an assessment of your actual server (*or hosting*) environment called the Cardholder Data Environment (*CDE*). Obtaining "PCI Compliance" is the responsibility of you the merchant and your hosting provider, working together, using PCI compliant architecture with proper hardware & software configurations and access control procedures.

The PA-DSS Validation is intended to ensure that OpenEPS will help you facilitate and maintain PCI Compliance with respect to how the payment application handles user accounts, passwords, encryption, and other payment data related information.

The Payment Card Industry (*PCI*) has developed security standards for handling cardholder information in a published standard called the PCI Data Security Standard (*DSS*). The security requirements defined in the DSS apply to all members, merchants, and service providers that store, process, or transmit cardholder data.

The PCI DSS requirements apply to all system components within the payment application environment which is defined as any network device, host, or application included in, or connected to, a network segment where cardholder data is stored, processed or transmitted.

Requirements of PCI DSS:

1. **Build and Maintain a Secure Network and Systems**
    a. Install and maintain a firewall configuration to protect cardholder data
    b. Do not use vendor-supplied defaults for system passwords and other security parameters
2. **Protect Cardholder Data**
    a. Protect stored cardholder data
    b. Encrypt transmission of cardholder data across open/public networks
3. **Maintain a Vulnerability Management Program**
    a. Protect all systems against malware and regularly update anti-virus software or programs

   b. Develop and maintain secure systems and applications

4. **Implement Strong Access Control Measures**
   a. Restrict access to cardholder data by business need-to-know
   b. Identify and authenticate access to system components
   c. Restrict physical access to cardholder data
5. **Regularly Monitor and Test Networks**
   a. Track and monitor all access to network resources and cardholder data
   b. Regularly test security systems and processes
6. **Maintain an Information Security Policy**
   a. Maintain a policy that addresses information security for all personnel

Implementation of Payment Application in a PCI-Compliant Environment

The following areas must be considered for proper implementation in a PCI-Compliant environment:

- Remove Historical Sensitive Authentication Data
- Handling of Sensitive Authentication Data
- Secure Deletion of Cardholder Data
- All PAN is masked by default
- Cardholder Data Encryption & Key Management
- Removal of Historical Cryptographic Material

Under the guidance of our QSA, the answers below does not include store and forward scenarios.

Remove Historical Sensitive Authentication Data (PA-DSS 1.1.4)

Sensitive Authentication Data includes security-related information (*including but not limited to card validation codes/values, full track data (from the magnetic stripe or equivalent on a chip*)), PINs, and PIN blocks used to authenticate cardholders and/or authorize payment card transactions. (*Sensitive Authentication Data as defined in the PCI SSC's Glossary of Terms, Abbreviations, and Acronyms*)

OpenEPS is designed not to allow for the retention of sensitive authentication data. Therefore, there is no need for secure deletion of this historical data by the application as required by PA-DSS v3.1.

Handling of Sensitive Authentication Data (PA-DSS 1.1.5)

OpenEPS does not store Sensitive Authentication Data for any reason after authorization. The only instance of storage is in store/forward situation in offline mode. After the encrypted blob (*transaction*) has been authorized the encrypted blob is over written. Here is how the process works:

- The only location that credit/debit card data (*prior to authorization*) is stored in the OpenEPS application is the in the offline file.
    - 3DES DUKPT – DEK (*3DES 168 bit encrypted message*) message
    - OpenEPS – no communication with Connected Payments
    - DEK message is encrypted with AES 256 and stored in the offline file (*DOUBLE ENCRYPTION*)
    - Store/forward – encrypted information is stored here until communications is restored with Connected Payments and OpenEPS continually try to send the offline messages up and then overwrites the encrypted data after it is sent with nulls, deletes the file, and creates a new offline file
    - OpenEPS does not store any credit/debit card data.
- Offlines continually attempted to forward encrypted transactions to Connected Payments. If there are offlines that do not forward, a threshold is set (*configurable*) to send an alert when the threshold has been reach that the offline transactions are not being forwarded.
- OpenEPS does not have access to any clear text PAN and the offline file is encrypted with AES 256.

Secure Deletion of Cardholder Data (PA-DSS 2.1)

OpenEPS does not store cardholder data (*PAN*) and therefore there is no data to be purged by the application as required by PA-DSS v3.1. To protect further against inadvertent storage of card holder data please see Addendum 1.

All PAN is Masked by Default (PA-DSS 2.2)

OpenEPS does not have the ability to display full PAN for any reason and therefore there is no configuration details to be provided as required for PA-DSS v3.1.

Cardholder Data Encryption & Key Management (PA-DSS 2.3, 2.4, and 2.5)

OpenEPS does not store cardholder data in any way nor does it provide any configurability that would allow a merchant to store cardholder data, therefore no encryption of cardholder data is required for PA-DSS v3.1.

Removal of Historical Cryptographic Material (PA-DSS 2.6)

OpenEPS using a key strategy where the DEK is a dynamic on time use key. A KEK is used to encrypt the DEK to send to Connected Payments. With each update, historical key parts are removed and replaced with a new key part. [*Does not apply to P2P*]

Set up Strong Access Controls (3.1 and 3.2)

The PCI DSS requires that access to all systems in the payment processing environment be protected through use of unique users and complex passwords. Unique user accounts indicate that every account used is associated with an individual user and/or process with no use of generic group accounts used by more than one user or process.

Authentication credentials are not generated or managed by OpenEPS. Instead, authentication credentials used by OpenEPS are provided by the underlying operating system or other authentication software such as active directory in your environment. To maintain PCI DSS compliance the following 11 points must be followed per the PCI DSS:

1. You must not use or require the use of default administrative accounts for other that necessary or required software (*e.g. database default administrative accounts*) (*PCI DSS 2.1 / PA-DSS 3.1.1*)
2. You must assign unique IDs for all user accounts. (*PCI DSS 8.1.1 / PA-DSS 3.1.3*)
3. You must provide at least one of the following three methods to authenticate users: (*PCI DSS 8.2 / PA-DSS 3.1.4*)
    a. Something you know, such as a password or passphrase
    b. Something you have, such as a token device or smart card
    c. Something you are, such as a biometric
4. You must NOT require or use any group, shared, or generic accounts and passwords (*PCI DSS 8.5 / PA-DSS 3.1.5*)
5. You must configure passwords must to be at least 7 characters and includes both numeric and alphabetic characters (*PCI DSS 8.2.3 / PA-DSS 3.1.6*)
    a. Passwords must not be short or simple to guess.
    b. Passwords must be difficult to guess.
    c. Passwords must not contain words that are in a dictionary, proper names, geographical locations, common acronyms, slang, derivatives of the user login ID, or common sequences such as “123456.”
    d. The user is must be discourage from choosing passwords that are associated with personal details of the user’s personal life, such as birthdays, wedding anniversaries, phone numbers, social security numbers, or other forms of identifiable information.
    e. Alternatively, the passwords/phrase must have complexity and strength at least equivalent to the parameters specified above.

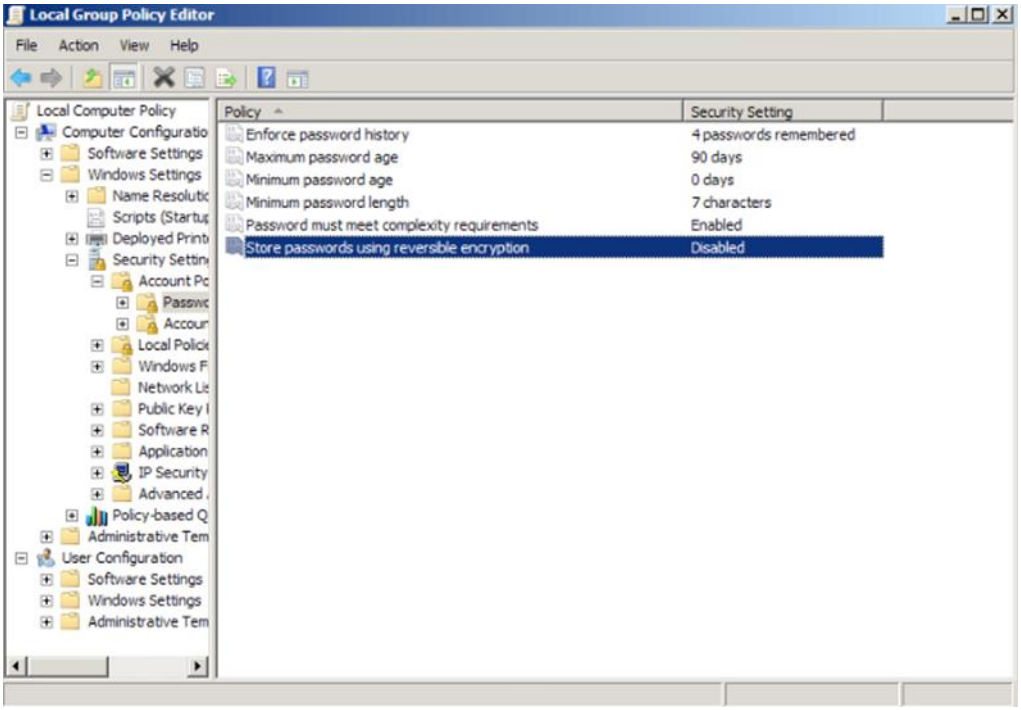
6. You must configure passwords to be changed at least every 90 days (*PCI DSS 8.2.4 / PA-DSS 3.1.7*)
    a. All users must be forced to change their password at least once every 90 days.
    b. If a password is not change, the user must have his/her access to the system denied.
    c. Only the system administrator will be able to restore the user's account.
    d. Additionally, if the user has shared his/her password with an unauthorized individual or the password has been compromised by no means of the user, the user is required to change his/her password.
7. You must configure passwords so that password history is kept and requires that a new password is different than any of the last four passwords used (*PCI DSS 8.2.5 / PA-DSS 3.1.8*)
8. The payment application limits repeated access attempts by locking out the user account after not more than six logon attempts (*PCI DSS 8.1.6 / PA-DSS 3.1.9*)
9. The payment application sets the lockout duration to a minimum of 30 minutes or until an administrator enables the user ID. (*PCI DSS 8.1.7 / PA-DSS 3.1.10*)
10. The payment application requires the user to re-authenticate to re-activate the session if the application session has been idle for more than 15 minutes. (*PCI DSS 8.1.8 / PA-DSS 3.1.11*)

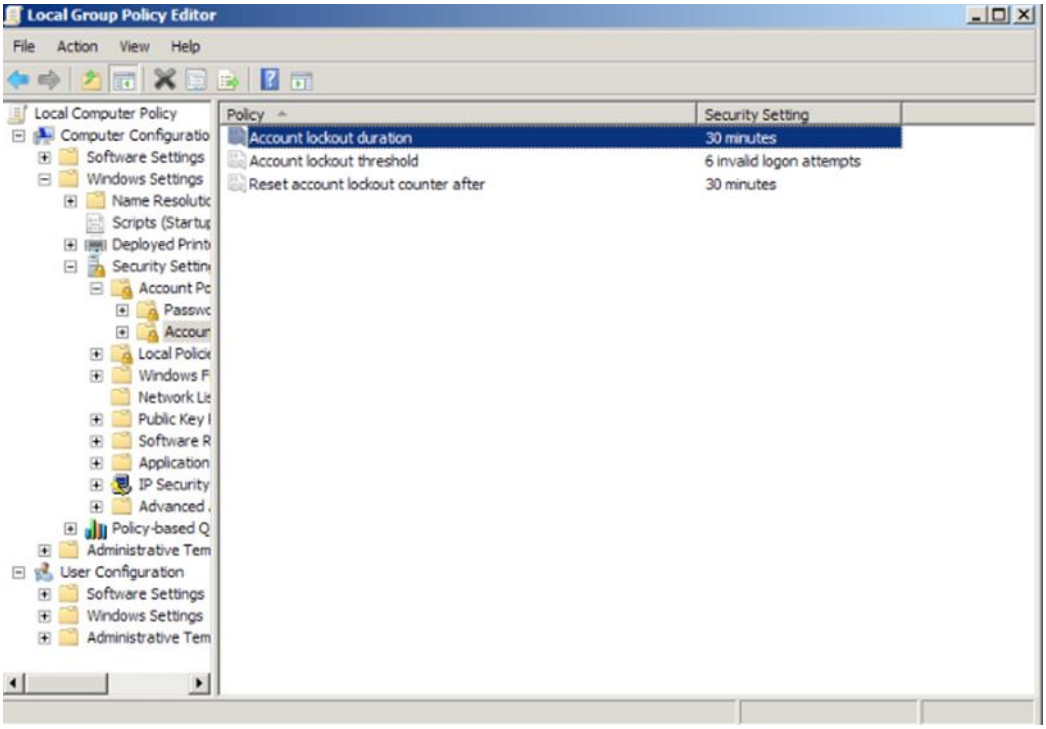
Here are steps that Microsoft provides to comply with PCI requirements:

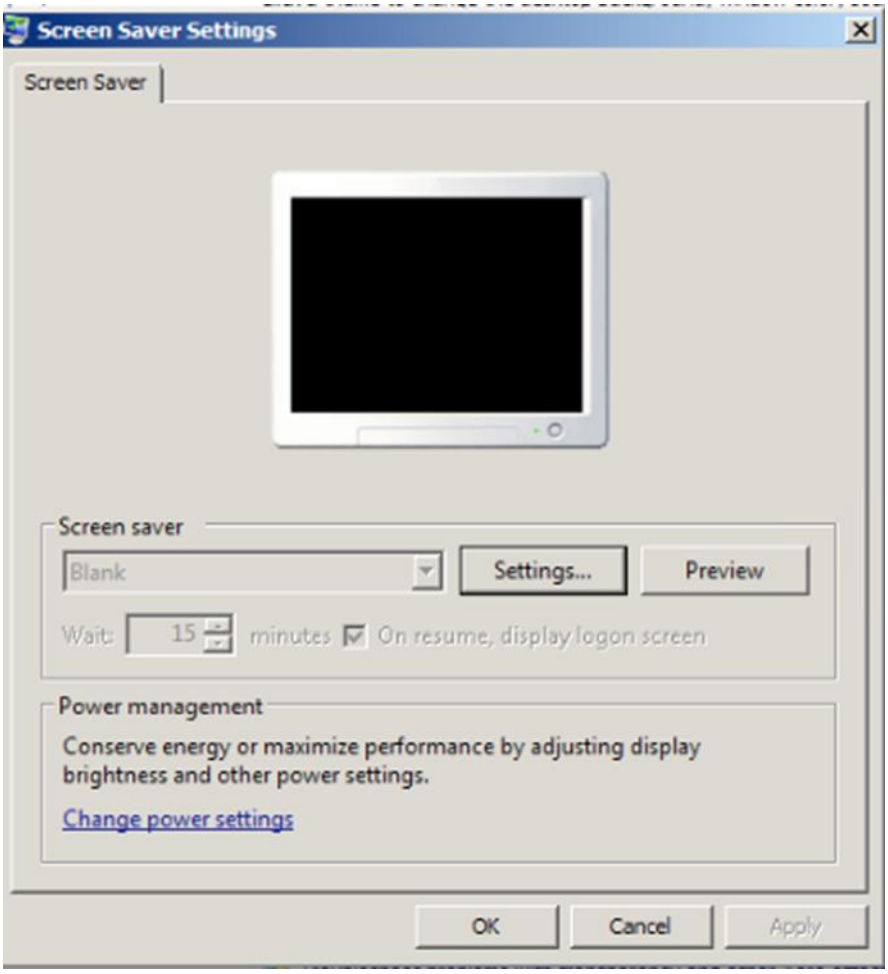
a. You must be logged on as an administrator to perform these steps.
b. If your computer is on a domain, only your network administrator can change password policy settings.
c. You can help protect your computer by customizing your password policy settings, including requiring users to change their password regularly, specifying a minimum length for passwords, and requiring passwords to meet certain complexity requirements.
d. Open Local Security Policy by clicking the Start button Picture of the Start button, typing secpol.msc into the search box, and then clicking secpol. Administrator permission required if you're prompted for an administrator password or confirmation, type the password or provide confirmation.
e. In the left pane, double-click Account Policies, and then click Password Policy.
f. Double-click the item in the Policy list that you want to change, change the setting, and then click OK.
    i. Policies are as follows:
        1. Enforce Password History
        2. Maximum Password Age
        3. Minimum Password Age
        4. Minimum Password length
        5. Password must meet complexity requirements
        6. Store passwords using reversible encryption

        (*http://windows.microsoft.com/en-us/windows/change-password-policy-settings#1TC=windows-7*)

Configure windows group policy and screensaver settings in the following manner:







You must assign strong passwords to any default accounts (*even if they won't be used*), and then disable or do not use the accounts. (*Note: These password controls are not intended to apply to employees who only have access to one card number at a time to facilitate a single transaction. These controls are applicable for access by employees with administrative capabilities, for access to systems with cardholder data, and for access controlled by the application. The requirements apply to the payment application and all associated tools used to view or access cardholder data.*) PA-DSS 3.2: Control access, via unique username and PCI DSS-compliant complex passwords, to any PCs or servers with payment applications and to databases storing cardholder data.

Properly Train and Monitor Admin Personnel

It is your responsibility to institute proper personnel management techniques for allowing admin user access to cardholder data, site data, et cetera. In most systems, a security breach is the result of unethical personnel. So pay special attention to whom you trust into your admin site and who you allow to view full decrypted and unmasked payment information. [*Note ~ OpenEPS is designed to prevent any access to PAN or other sensitive card holder data information*]

Log settings must be compliant (PA-DSS 4.1.b, 4.4.b)

OpenEPS has PA-DSS compliant logging enabled by default. This logging is not configurable and may not be disabled. Disabling or subverting the logging function of OpenEPS in any way will result in non-compliance with PCI DSS.

OpenEPS provides logs that can either be stored on each POS lane or in a centralized location. These logs are provided by way of a text file that can be utilize for any logging system. The merchant can choose the duration to retain the logs. The merchants are responsible for implementing centralized logging, describe below the required PCI DSS-compliant log settings, per PCI Data Security Standard 10.2 and 10.3.

If the merchant choose to utilize centralize logging, the logs for each lane can be directed to that centralized server. Logging is via industry standard log file mechanisms such as text file. This text file can be imported into any centralize logging system.

NCR recommends that merchants follow the required PCI requirements for proper logging:

- Implement automated assessment trails for all system components to reconstruct the following events:
    - 10.2.1 All individual user who accesses cardholder data from the application
        - This does not apply in the case of OpenEPS. OpenEPS does not provide the ability for an individual to access card holder data from the application.
    - 10.2.2 All actions taken by any individual with administrative privileges in the application
        - These individuals should be kept to a minimum and only provide enough rights and privileges to perform his/her function required by his/her job.
    - 10.2.3 Access to application audit trails managed by or within the application
        - This would not apply to OpenEPS and only applies to the merchant's application that monitors OpenEPS's logs.
    - 10.2.4 Invalid logical access attempts
        - This should apply to all access to the POS.
    - 10.2.5 Use of the application's identification and authentication mechanisms (*including but not limited to creation of new accounts, elevation of privileges, et cetera*) and all changes, additions, deletions to application accounts with root or administrative privileges
        - This does not apply to OpenEPS, however, does apply to the POS and the application that is correlating the logs. These events should be strictly adhered.
    - 10.2.6 Initialization, stopping, or pausing of the application audit logs
        - These events must be recorded.
    - 10.2.7 Creation and deletion of system-level objects within or by the application
        - This would apply only to the POS and not OpenEPS.

- Record at least the following assessment trail entries for all system components for each event from 10.2.x above:
    - 10.3.1 User identification
    - 10.3.2 Type of event
    - 10.3.3 Date and time
    - 10.3.4 Success or failure indication
    - 10.3.5 Origination of event
    - 10.3.6 Identity or name of affected data, system component, or resource.

NCR also recommends that audit policies to individual files and folders on your POS have permissions set to record successful access attempts or failed access attempts in the security log. See Addendum 2 for a sample of how to audit polices for files and folders on a Windows 7 system.

PCI-Compliant Wireless settings (PA-DSS 6.1.a and 6.2.b)

OpenEPS does not support wireless technologies. However, should the merchant implement wireless access within the cardholder data environment, the following guidelines for secure wireless settings must be followed per PCI Data Security Standard 1.2.3, 2.1.1 and 4.1.1:2.1.1: Change wireless vendor defaults per the following 5 points:

1. Encryption keys must be changed from default at installation, and must be changed anytime anyone with knowledge of the keys leaves the company or changes positions. Changing of these keys must include all access points, controllers, routers, any access into the wireless network that includes user credentials, VPNs, WPA2 keys, IPsec, et cetera.
2. Default SNMP community strings on wireless devices must be changed. For example on a Cisco controller:
    a. Using the CLI to Change the SNMP Community String Default Values
        1. To see the current list of SNMP communities for this controller, enter this command: show snmp community
        2. If "public" or "private" appears in the SNMP Community Name column, enter this command to delete this community: config snmp community delete name
            1. The name parameter is the community name (in this case, "public" or "private").
        3. To create a new community, enter this command: config snmp community create name
            1. Enter up to 16 alphanumeric characters for the name parameter. Do not enter "public" or "private."
        4. To enter the IP address from which this device accepts SNMP packets with the associated community, enter this command: config snmp community ipaddr ip_address ip_mask name

5. To specify the access level for this community, enter this command, where ro is read-only mode and rw is read/write mode: config snmp community accessmode {ro | rw} name
6. To enable or disable this SNMP community, enter this command: config snmp community mode {enable | disable} name
7. To save your changes, enter save config.
8. Repeat this procedure if you still need to change the default values for a "public" or "private" community string.

b. Default passwords/passphrases on access points must be changed.

1. configure terminal (*enter global configuration mode*)
2. enable password (*enter a new password here*)
3. end (*return to privilege exec mode*)
4. show running-config (*verify entry*)
5. copy running-config startup-config (*saves entry to configuration file*)

c. Firmware on wireless devices must be updated to support strong encryption for authentication and transmission over wireless networks.

d. Other security-related wireless vendor defaults, if applicable, must be changed

3. Perimeter firewalls must be installed between any wireless networks and systems that store cardholder data, and these firewalls must deny or control (*if such traffic is necessary for business purposes*) any traffic from the wireless environment into the cardholder data environment.
4. Industry best practices (*for example, IEEE 802.11.i*) must be used to implement strong encryption for authentication and transmission of cardholder data.
5. Note: The use of WEP as a security control was prohibited as of June 30, 2010.

Services and Protocols (PA-DSS 8.2.c)

OpenEPS does not require the use of any insecure services or protocols. Here are the services and protocols that OpenEPS does require:

- Message encryption, sFTP, HTTPS, IPSec, et cetera will require the following protocols and ciphers
  - TLS 1.2
  - RSA 2048 (*min*) or ECC 224 (*min*)
  - SHA2
  - AES (*GCM mode*)

Disabling CRL Checking on public networks for P2PE

NCR has become aware of issues with the TLS process for legacy POS. NCR contacted Coalfire, its QSA, to clarify the TLS process with specific regards to certification validation process. NCR is required to have TLS1.2, strong ciphers, and strong certificate validation for its certifications; PCI PA-DSS 3.1 and PCI DSS

3.1. These are requirements for NCR's certifications and that means that even though the 3DES DUKPT encryption is a possible compensating control when dealing with weak TLS issues, NCR must require that all customers utilize TLS 1.2 and the strong ciphers when accessing NCR's Cloud from the Internet.

With regards to our customers that are using legacy POS and are having issues with latency during NCR's certificate validation process, customers can turnoff certificate revocation and use the P2PE solution as a compensating control due to the strong encryption when utilizing 3DES DUKPT at the pin pad and still remain complaint with PCI DSS.

Finally, this is only the joint opinion of NCR and Coalfire. This P2PE 3DES DUKPT encryption process must be evaluated as a compensating control with your QSA to ensure your QSA agrees with this opinion prior to disabling CRL checking. Further, this is not applicable for any clients running any version of NCR's software encryption. Please see Addendum 3 for our QSA's letter regarding disabling CRL checking on public networks for OpenEPS versions using NCR's P2PE solution.

Never store cardholder data on internet-accessible systems (PA-DSS 9.1.c)

Never store cardholder data on Internet-accessible systems (*e.g., web server and database server must not be on the same server.*) OpenEPS is design not to store cardholder data in a client's environment.

PCI-Compliant Remote Access (10.1)

The PCI standard requires that if employees, administrators, or vendors are granted remote access to the payment processing environment; access should be authenticated using a two-factor authentication mechanism. The means two of the following three authentication methods must be used:

1. Something you know, such as a password or passphrase
2. Something you have, such as a token device or smart card
3. Something you are, such as a biometric

NCR team members do not have access into merchant's environment. Connected Payments cloud environment utilizes two-factor authentication and this is outside the scope of the merchant's environment.

PCI-Compliant Delivery of Updates (PA-DSS 10.2.1.a)

OpenEPS delivers patches and updates in a secure manner through an automated process using message encryption (*TLS 1.2*).

- OpenEPS is actively in development and deploys patches and updates based upon criticality of the bug or security vulnerability.
    - There are two major releases per year of OpenEPS

    - Critical bug fixes and security vulnerabilities are release as quickly as possible. The intervals may be within weeks/months of the discovery of the issue or in the next release; whichever is sooner.
  - OpenEPS patches and updates are pushed by NCR automated processes or through manual download by customers in their environments. Delivery is in a secure manner with a known strong TLS authentication and encryption and at NCR's cloud. OpenEPS is design only to accept patches and updates from NCR's Connected Payment's cloud.
- For manual updates, delivery is through strong TLS at NCR's website or through NCR's sFTP server.
- OpenEPS utilizes a SHA2 hash verification with NCR's Connected Payment's cloud. Connected Payment's cloud compares the hash of what OpenEPS generates and what is stored in the cloud. If the two do not match, OpenEPS will not process transactions.

As NCR is a development company, we keep abreast of the relevant security concerns and vulnerabilities in our area of development and expertise. We do this by:

- Attending security conferences and training
- Subscribed to organizations such as the following:
    - SANS Top 25
    - Infraguard (*FBI*)
    - U.S. Certs – National Cyber Awareness
    - FS-ISAC (*Financial Service – Information Sharing and Analysis Center*)
    - Owasp Top 10

Once we identify a relevant vulnerability, we work to develop & test a patch that helps protect OpenEPS against the specific, new vulnerability. We attempt to publish a patch within 30 days or sooner of the identification of the vulnerability. We will then contact vendors and dealers to encourage them to install the patch. Typically, merchants are expected to respond quickly to and install available patches within 30 days.

We do not deliver software and/or updates via remote access to customer networks. OpenEPS patches and updates are pushed by NCR automated processes or through manual download by customers in their environments.

PCI-Compliant Remote Access (10.2.3.a)

NCR team members do not have remote access into a merchant's environment. This section does not apply to OpenEPS. However, NCR has been instructed to provide this information for merchants whom decide to implement remote access into their environments.

PCI requires the following requirements if employees, administrators, or vendors are granted remote access to the merchant's environment.

- Access to be granted using two-factor authentication

- For vendor remote access, vendor remote access accounts should only be active when access is required into the merchant's environment. Access should be limited to the rights required for the vendor to perform under contract.
- All remote communications sessions must meet strong TLS, VPN, or SSH standards as required by PCI DSS 3.1 and PA DSS 12.1.
- Default credentials and setting are required to be changed for any remote application or device.
- Connections should be restricted to the remote device that has been previously authorized.
- Use strong authentication and complex passwords for logins according to PA-DSS 3.1.1 – 3.1.10 and PCI DSS 8.1, 8.3, and 8.5.8-8.5.15
- Only allow remote access through VPN that routes through a firewall; no direct Internet connections allowed.
- Logging must be enable for auditing.
- Passwords must be established according to PA-DSS 3.1.1 – 3.1.10 and PCI DSS Requirements 8.1, 8.2, 8.4, and 8.5.

Data Transport Encryption (PA-DSS 11.1.b)

The PCI DSS requires the use of strong cryptography and encryption techniques with at least a 128 bit encryption strength (*either at the transport layer with TLS or IPSEC; or at the data layer with algorithms such as RSA or AES*) to safeguard cardholder data during transmission over public networks (*this includes the Internet and Internet accessible DMZ network segments*).

PCI DSS requirement 4.1: Use strong cryptography and security protocols such as transport layer security (*TLS 1.2*) and Internet protocol security (*IPSEC*) to safeguard sensitive cardholder data during transmission over open, public networks.

Examples of open, public networks that are in scope of the PCI DSS are:

- The Internet
- Wireless technologies
- Global System for Mobile Communications (*GSM*)
- General Packet Radio Service (*GPRS*)

Refer to the Dataflow diagrams on page 13 and 14 for an understanding of the flow of encrypted data associated with OpenEPS.

To lessen the dependency on the OS for PCI compliancy, NCR has modified OpenEPS to embed strong security ciphers within our solution, including TLS1.2, RSA 2048 or ECC 224, SHA2, and AES (*GCM mode*). With these changes, OpenEPS will no longer be dependent on the OS to provide strong protocol encryption. NCR recommends that retailers update to these latest version of OpenEPS in order to provide the strongest encryption and to be prepared for PCI 3.1 compliancy.

PCI-Compliant Use of End User Messaging Technologies (PA-DSS 11.2.b)

OpenEPS does not allow or facilitate the sending of PANs via any end user messaging technology (*for example, e-mail, instant messaging, and chat*).

Non-console administration (PA-DSS 12.1)

Although OpenEPS does not support non-console administration and we do not recommend using non-console administration, should you ever choose to do this, you must use SSH, VPN, or TLS 1.2 or higher for encryption of this non-console administrative access.

Network Segmentation

The PCI DSS requires that firewall services be used (*NAT or PAT*) to segment network segments into logical security domains based on the environmental needs for internet access. Traditionally, this corresponds to the creation of at least a DMZ and a trusted network segment where only authorized, business-justified traffic from the DMZ is allowed to connect to the trusted segment. No direct incoming internet traffic to the trusted application environment can be allowed. Additionally, outbound internet access from the trusted segment must be limited to required and justified ports and services.

Refer to the standardized Network diagram on page 12 for an understanding of the flow of encrypted data associated with OpenEPS.

Maintain an Information Security Program

In addition to the preceding security recommendations, a comprehensive approach to assessing and maintaining the security compliance of the payment application environment is necessary to protect the organization and sensitive cardholder data.

The following is a very basic plan every merchant should adopt in developing and implementing a security policy and program:

- Read the PCI DSS in full and perform a security gap analysis. Identify any gaps between existing practices in your organization and those outlined by the PCI requirements.
- Once the gaps are identified, determine the steps to close the gaps and protect cardholder data. Changes could mean adding new technologies to shore up firewall and perimeter controls, or increasing the logging and archiving procedures associated with transaction data.
- Create an action plan for on-going compliance and assessment.
- Implement, monitor and maintain the plan. Compliance is not a one-time event. Regardless of merchant or service provider level, all entities should complete annual self-assessments using the PCI Self-Assessment Questionnaire.
- Call in outside experts as needed.

Application System Configuration

Below are the operating systems and dependent application patch levels and configurations supported and tested for continued PCI DSS compliance.

Hardware Requirements for OpenEPS

- Pentium 4 (*Intel or compatible*) 500 MHz processor (*1 GHz or faster recommended*)
  - The processor must contain instructions to support strong cryptography
- 1 GB of RAM (*or more recommended*)
- VGA, or higher, resolution monitor set at 1024x768 or better
- Ethernet Card
- Drive Space Requirement for OpenEPS on each POS lane:
- 100 Mb free drive space for configuration files and logs

Software Requirements for OpenEPS (*Front POS Lanes*):

- TCP/IP Protocol
- Any of the following Operating Systems:
  - POSReady 7
  - POSReady 2009 (*with extended support*)
  - Windows 7
  - Windows 8

Payment Application Initial Setup & Configuration

NCR provides training for OpenEPS PCI compliant installation. This comprehensive training covers installation, configuration, and PCI regulations that affect the installation for the OpenEPS client.

If you wish to take part in one of our training sessions, contact the NCR division using the e-mail or telephone number in the Contact Information section above.

The content of this training is updated annually, required by PCI, or product enhancements, so even if you have taken the training before, you may wish to enroll in the latest session in order to acquire information on any of the new changes to the OpenEPS.

OpenEPS installation and configuration is as follows:

Double click on the OpenEPS executable. OpenEPS executable will install the DLL in the appropriate location. The OpenEPS executed will ask the following questions and the information must be supplied at that time:

- What company and store number?
- What hard drive?

The next steps to complete the installation is by clicking next to finish the installation.

OpenEPS DLL is installed in the default directory:

- C:\Program Files\MicroTrax\OpenEPS\

In addition it is highly recommended that the OpenEPS directory be protected through the use of a File Integrity Monitoring System. OpenEPS directories contain configuration information that could potentially be altered with malicious intent. Specific vulnerable files are the host files and the Setup.Txt, as these contain the IP addresses in use and could be manipulated potentially redirect payment processing traffic.

When using a File Integrity Monitoring System, be aware that certain files (*typically log or database files: *.tor, Spool*, actlog*, jrnl*, Offlines*) are constantly changing. It is often useful to either exclude these files from alerts completely, or configure the alerting software to allow the OpenEPS software to freely manipulate files within its directory structure, and to configure alerts for when files are directly manipulated by user accounts or when manipulated by other software. All directories listed must deny access to non-administrative users and be monitored by a File Integrity Monitoring System.

POS software must have read/write permissions to the OpenEPS directory. This is because OpenEPS is a DLL which the POS software loads, and therefor OpenEPS derives its permissions from the user account the POS is started under.

It is important to note, however, that the cashier, or other users of the POS must NOT have access to the OpenEPS folder. It is important for security that the casher or other daily users do not have the ability to modify OpenEPS configuration files. This generally requires that the POS be run under a separate specific user account, which is different from the user account actually used to log into the system by the cashier.

Recommend access rights for the OpenEPS folder:

- Admins: read/write
- POS account: read write
- Cashier: no access
- All others: no access

OpenEPS is installed in the following local registry directories:

- "HKEY_LOCAL_MACHINE\SOFTWARE\MTXEPS\OpenEPS\" or
- "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\MTXEPS\OpenEPS"

OpenEPS writes information to the Windows Registry locations noted above; specifically to store transaction data to file.

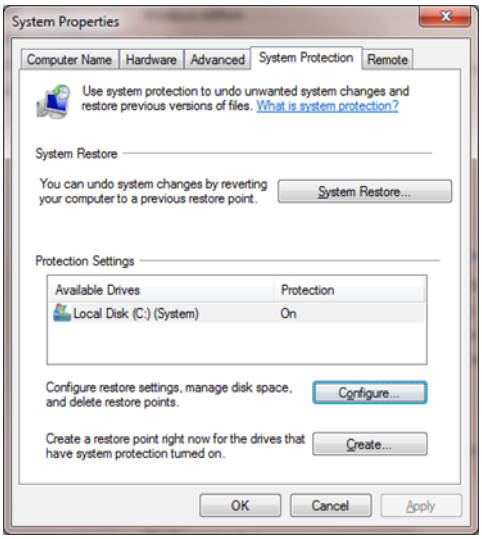
As such, the registry must be accessible to OpenEPS, however the registry keys noted above should not be accessible to any non-administrative user account. The registry key can be protected by limiting the permissions to the OpenEPS key to only those Windows accounts that require access.

ADDENDUM 1

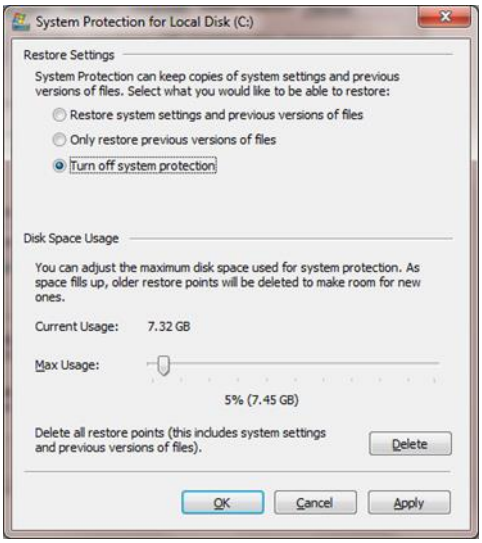
Addressing Inadvertent Capture of PAN

Disable System Restore Settings

- Disabling System Restore – Windows 7
  - Right Click on Computer > Select “Properties”
  - Select “System Protection” on the top left list, the following screen will appear:



- Select Configure, the following screen will appear:



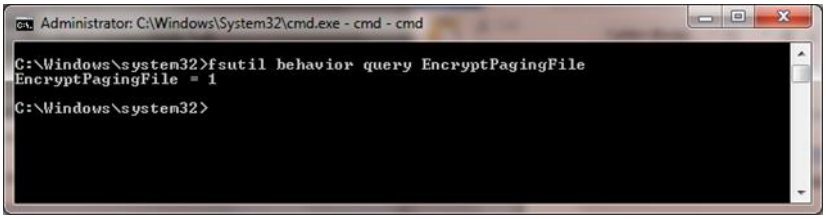
- Select “Turn off system protection”
- Click apply, and OK to shut the System Protection window
- Click OK again to shut the System Properties window
- Reboot the computer

Encrypt the System PageFile.sys

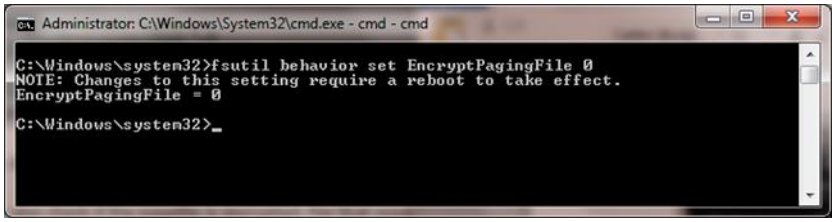
- Encrypting PageFile.sys – Windows 7

  * Please note that in order to perform this operation the hard disk must be formatted using NTFS.
- Click on the Windows “Orb” and in the search box type in “cmd”.
- Right click on cmd.exe and select “Run as Administrator”
- To Encrypt the Pagefile type the following command: fsutil behavior set EncryptPagingFile 1

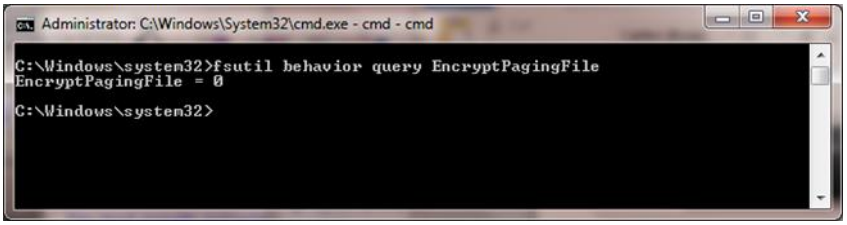
- To verify configuration type the following command: fsutil behavior query EncryptPagingFile



- If encryption is enabled EncryptPagingFile = 1 should appear
- In the event you need to disable PageFile encryption type the following command: fsutil behavior set EncryptPagingFile 0



- To verify configuration type the following command: fsutil behavior query EncryptPagingFile



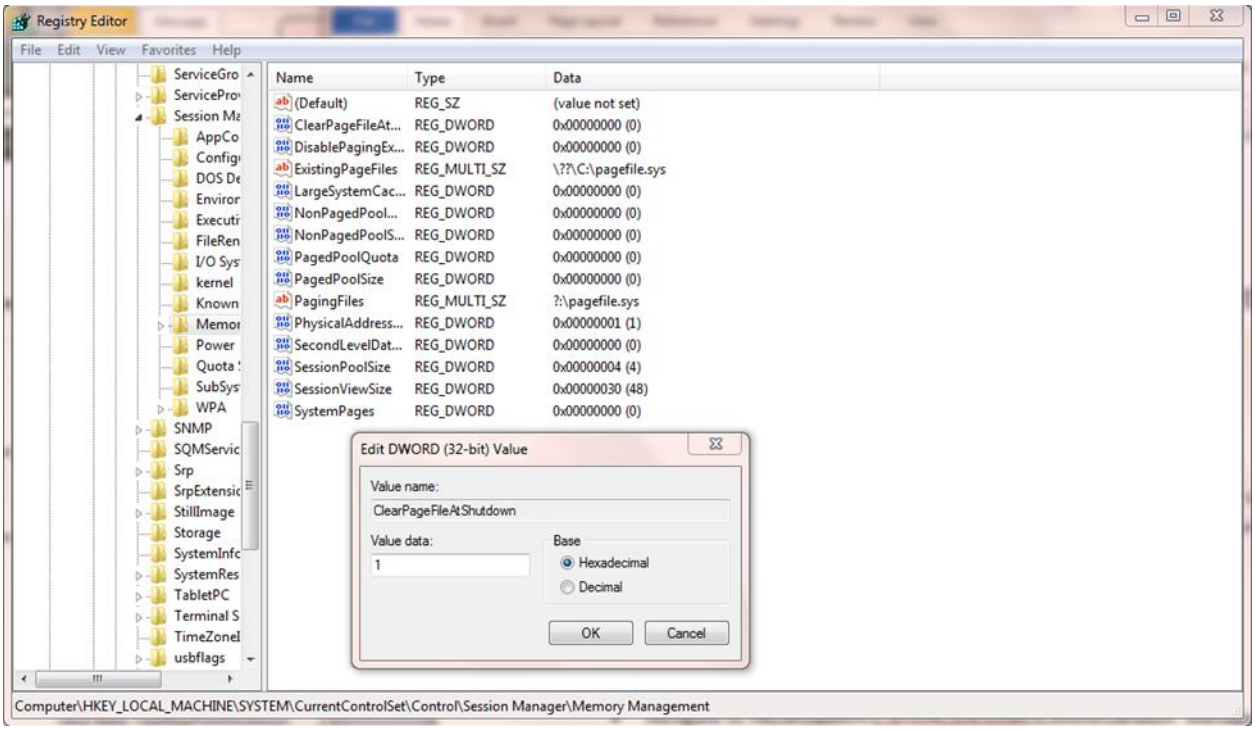
- If encryption is disabled EncryptPagingFile = 0 should appear

Clear the System Pagefile.sys on shutdown

Windows has the ability to clear the Pagefile.sys upon system shutdown. This will purge all temporary data from the pagefile.sys (*temporary data may include system and application passwords, cardholder data (PAN/Track), et cetera*).

NOTE: Enabling this feature may increase windows shutdown time.Click on the Windows “Orb” and in the search box type in “regedit”.

- Right click on regedit.exe and select “Run as Administrator”
- Navigate to HKLM\System\CurrentControlSet\Control\Session Manager\Memory Management
- Change the value from 0 to 1
- Click OK and close Regedit

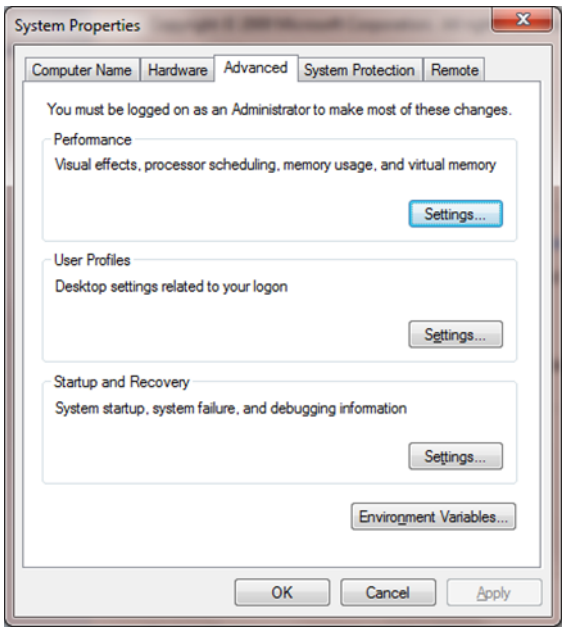


- If the value does not exist, add the following:
  - Value Name: ClearPageFileAtShutdown
  - Value Type: REG_DWORD
  - Value: 1

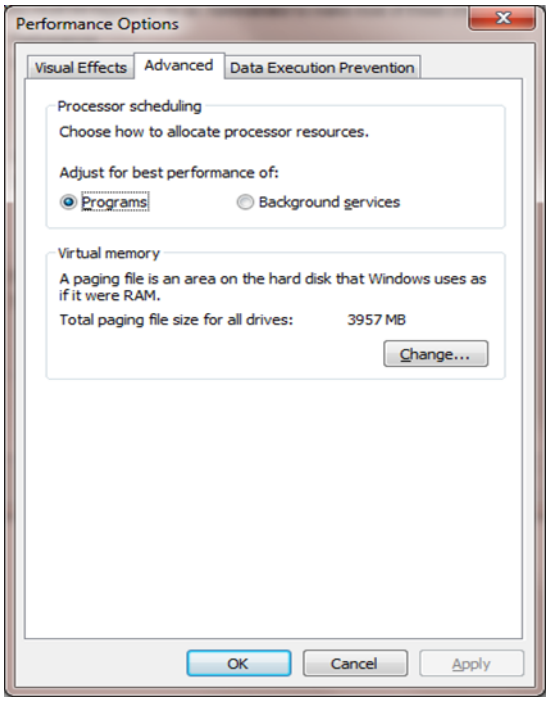
Disable System Management of Pagefile.sys

Disabling System Management of PageFile.sys – Windows 7

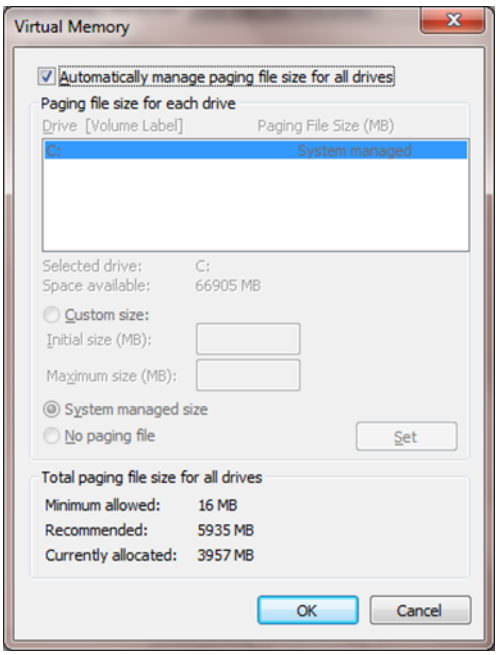
- Right Click on Computer > Select “Properties”
- Select “Advanced System Settings” on the top left list, the following screen will appear:



- Under performance select “Settings” and go to the “Advanced” tab, the following screen will appear:



- Select “Change” under Virtual Memory, the following screen will appear:

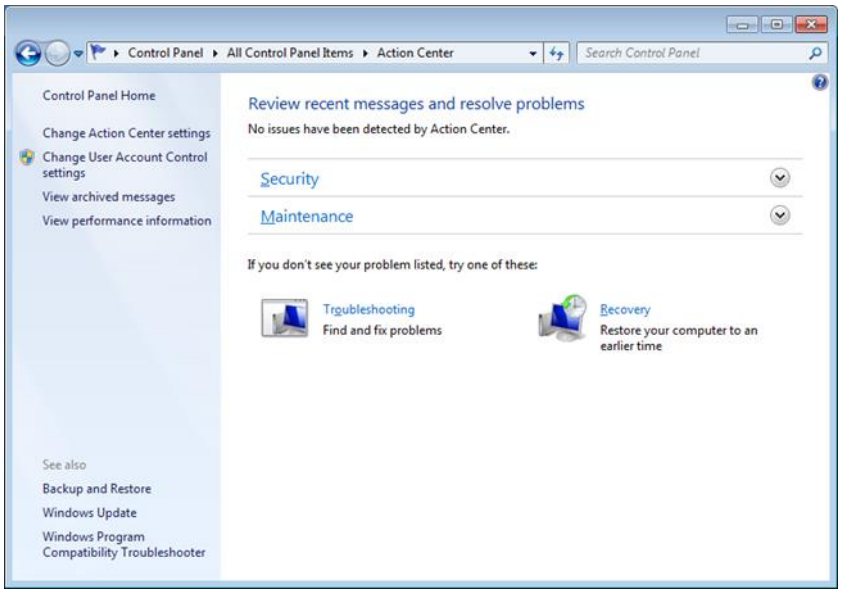


- Uncheck “Automatically manage page file size for all drives”
- Select “Custom Size”
- Enter the following for the size selections:
    - Initial Size – as a good rule of thumb, the size should be equivalent to the amount of memory in the system.
    - Maximum Size – as a good rule of thumb, the size should be equivalent to 2x the amount of memory in the system.
- Click “Ok”, “OK”, and “OK”
- You will be prompted to reboot your computer.

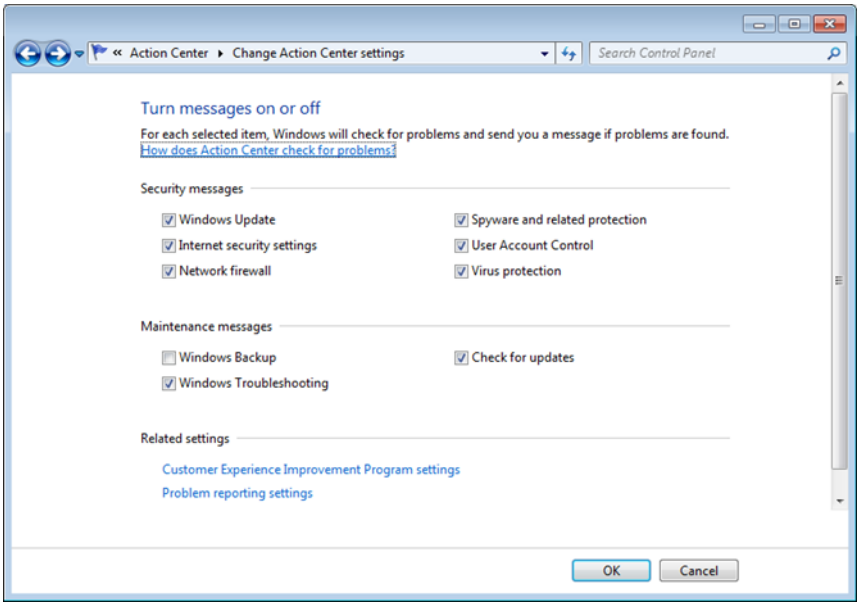
Disable Windows Error Reporting

Disabling Windows Error Reporting – Windows 7

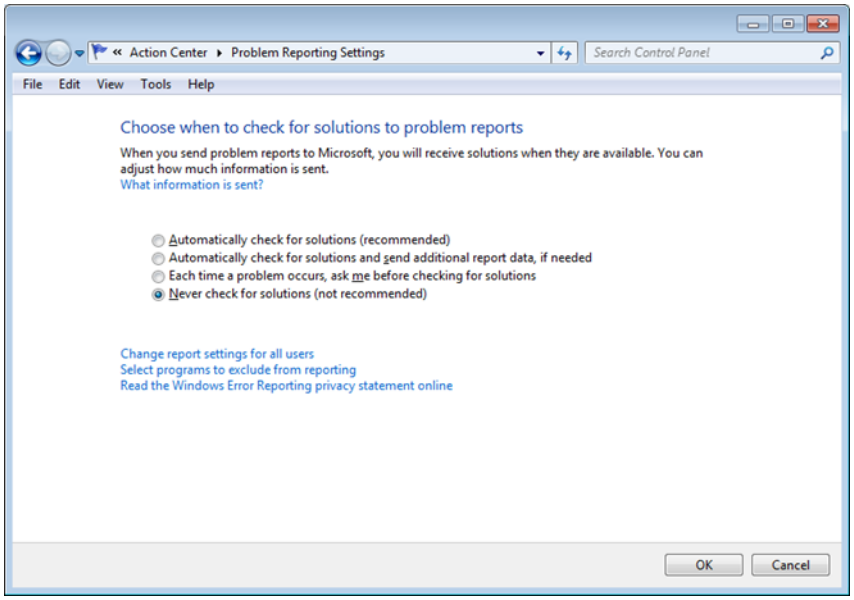
- Open the Control Panel
- Open the Action Center
- Select “Change Action Center Settings”



- Select “Problem Reporting Settings”



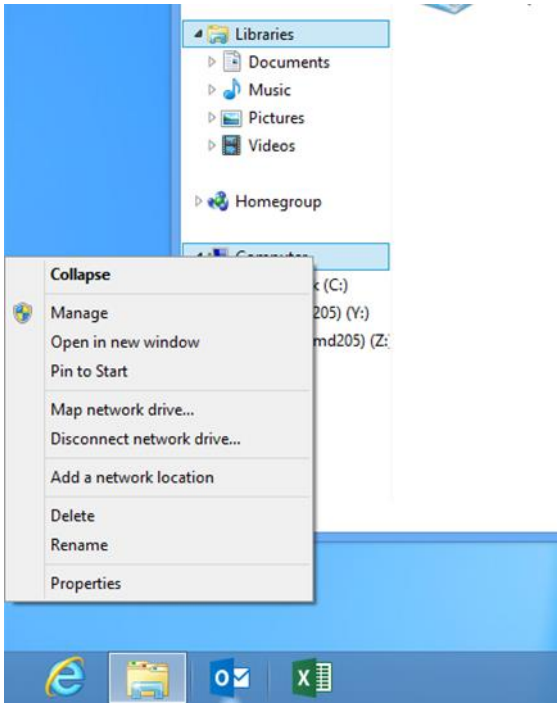
- Select “Never Check for Solutions”



Addressing Inadvertent Capture of PAN on WINDOWS 8

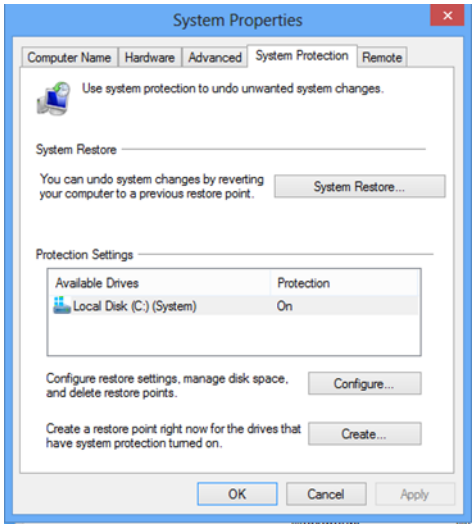
Disabling System Restore – Windows 8

- Right Click on Computer > Select “Properties”:

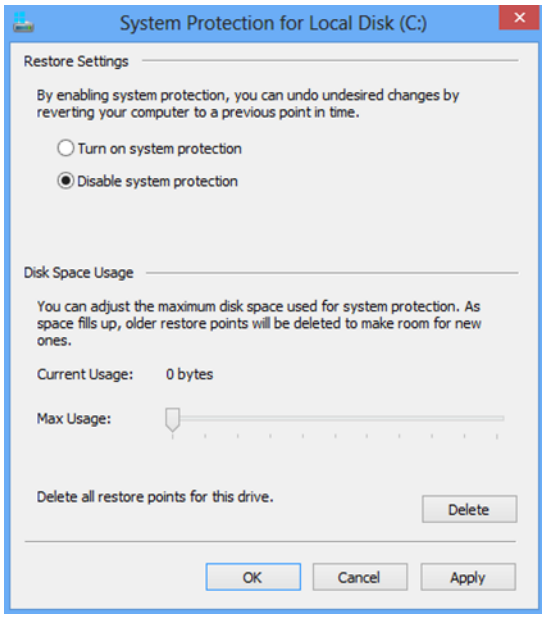


- Select “Advanced System Settings” from the System screen:

- Select “System Protection” on the top left list, the following screen will appear:



- Select Configure, the following screen will appear:



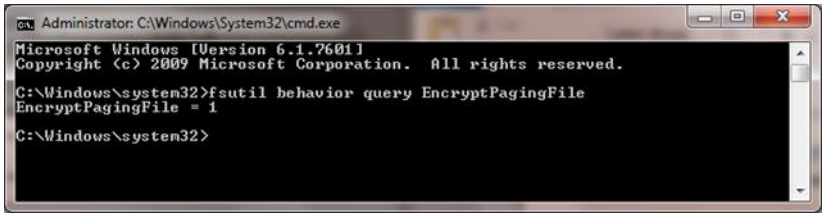
- Select “Disable system protection”
- Click apply, and OK to shut the System Protection window
- Click OK again to shut the System Properties window
- Reboot the computer

Encrypting PageFile.sys – Windows 8

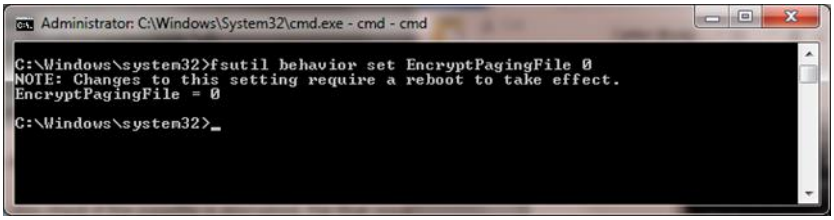
* Please note that in order to perform this operation the hard disk must be formatted using NTFS.

- From the desktop hold down the “Windows” key and type “F” to bring up the “Search” charm, select “Apps” in the “Apps” box type in “cmd”.
- Right click on “Command Prompt” icon located on the left side of your screen, a selection bar will appear at the bottom of the screen, select “Run as Administrator”
- To verify configuration type the following command: fsutil behavior query EncryptPagingFile”

- If encryption is enabled EncryptPagingFile = 1 should appear
- If encryption is disabled EncryptPagingFile = 0 should appear
- To Encrypt the Pagefile type the following command: fsutil behavior set EncryptPagingFile 1



- In the event you need to disable PageFile encryption type the following command: fsutil behavior set EncryptPagingFile 0

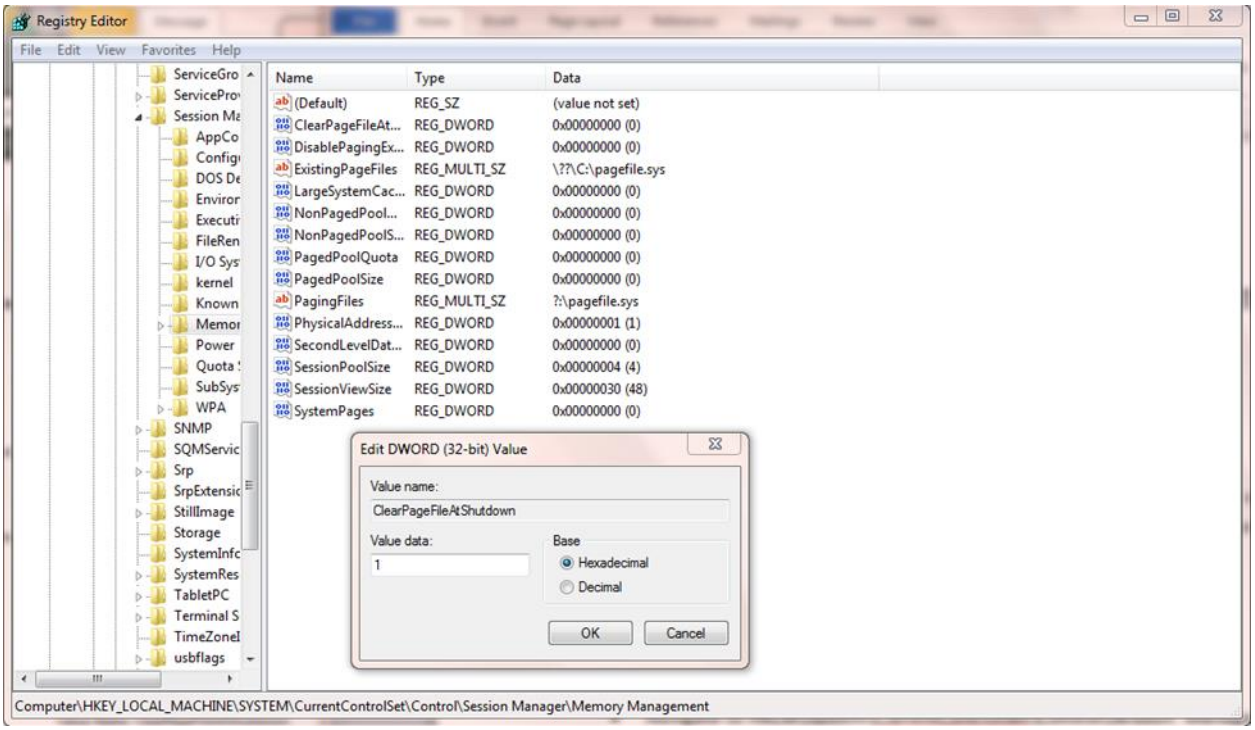


Clear the System Pagefile.sys on shutdown

Windows has the ability to clear the Pagefile.sys upon system shutdown. This will purge all temporary data from the pagefile.sys (*temporary data may include system and application passwords, cardholder data (PAN/Track), et cetera*).

NOTE: Enabling this feature may increase windows shutdown time.

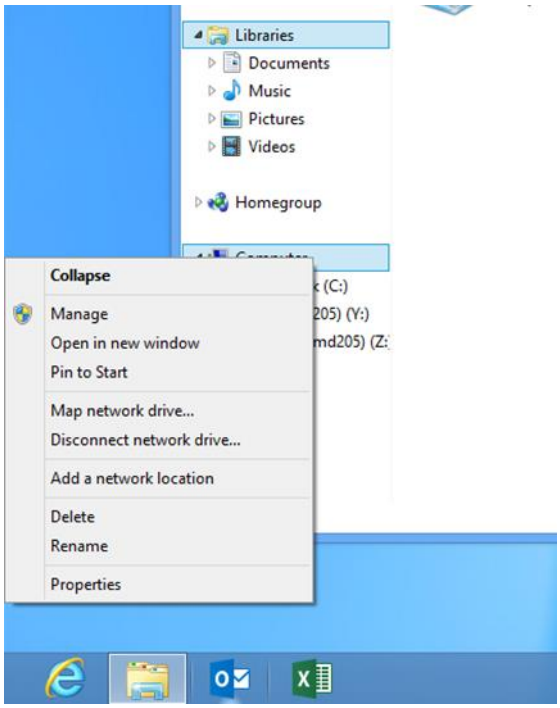
- From the desktop hold down the “Windows” key and type “F” to bring up the “Search” charm, select “Apps” in the “Apps” box type in “regedit”.
- Right click on regedit.exe and select “Run as Administrator”
- Navigate to HKLM\System\CurrentControlSet\Control\Session Manager\Memory Management
- Change the value from 0 to 1 on the “ClearPageFileAtShutdown” DWORD.
- Click OK and close Regedit



- If the value does not exist, add the following:
  - Value Name: ClearPageFileAtShutdown
  - Value Type: REG_DWORD
  - Value: 1

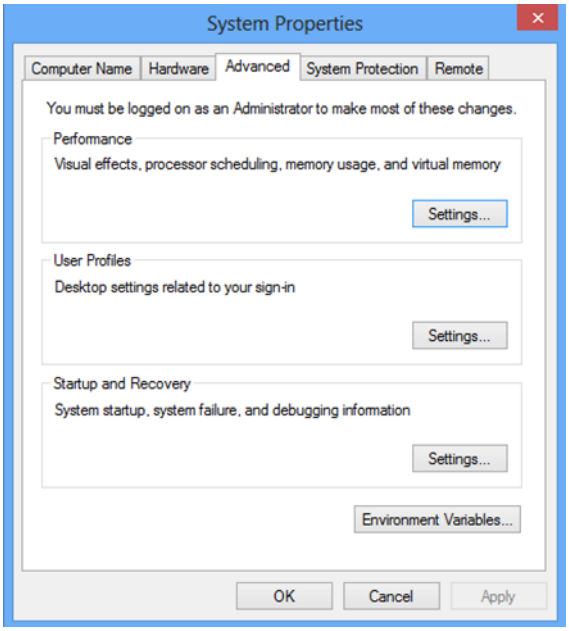
Disabling System Management of PageFile.sys – Windows 8

- Right Click on Computer > Select “Properties”:

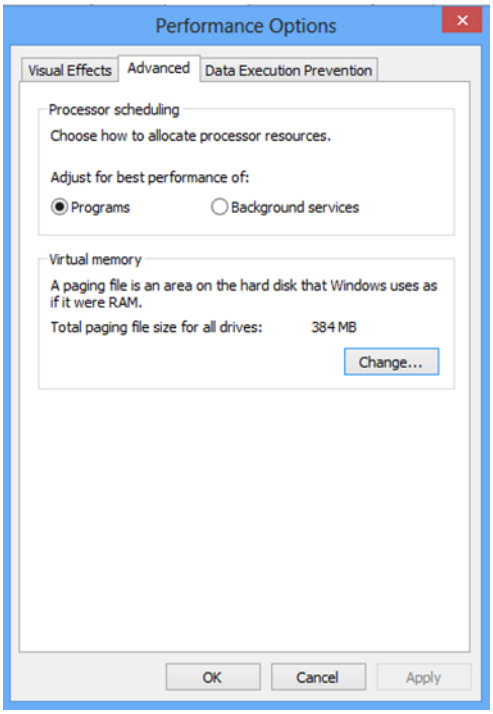


- Select “Advanced System Settings” from the System screen:

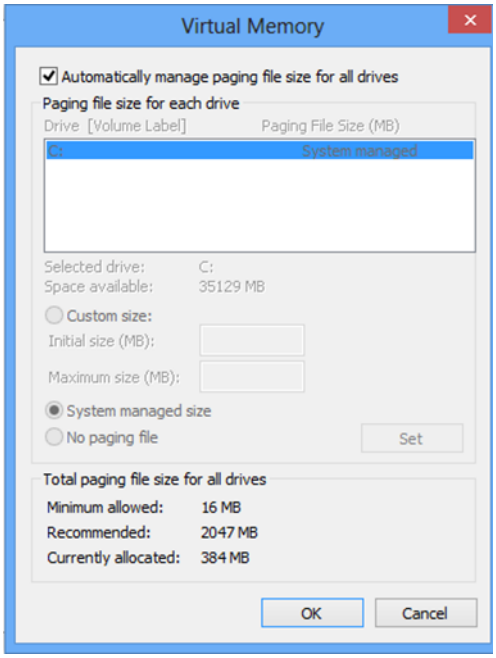
- Select the “Advanced” tab:



- Under performance select “Settings” and go to the “Advanced” tab, the following screen will appear:



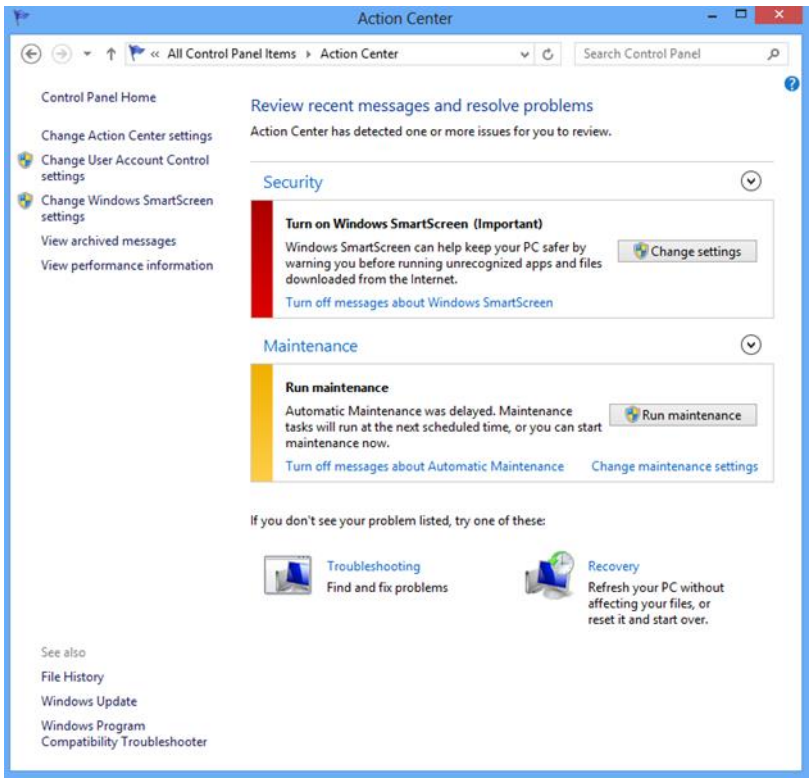
- Select “Change” under Virtual Memory, the following screen will appear:



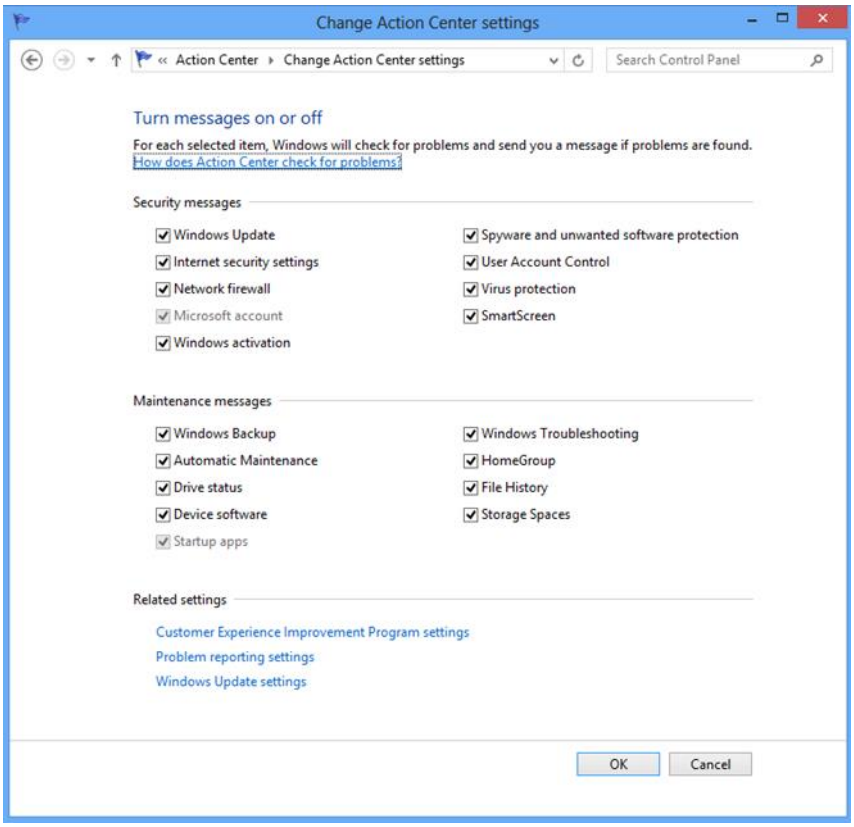
- Uncheck “Automatically manage page file size for all drives”
- Select “Custom Size”
- Enter the following for the size selections:
    - Initial Size – as a good rule of thumb, the size should be equivalent to the amount of memory in the system.
    - Maximum Size – as a good rule of thumb, the size should be equivalent to 2x the amount of memory in the system.
- Click “Ok”, “OK”, and “OK”
- You will be prompted to reboot your computer.

Disabling Windows Error Reporting – Windows 8

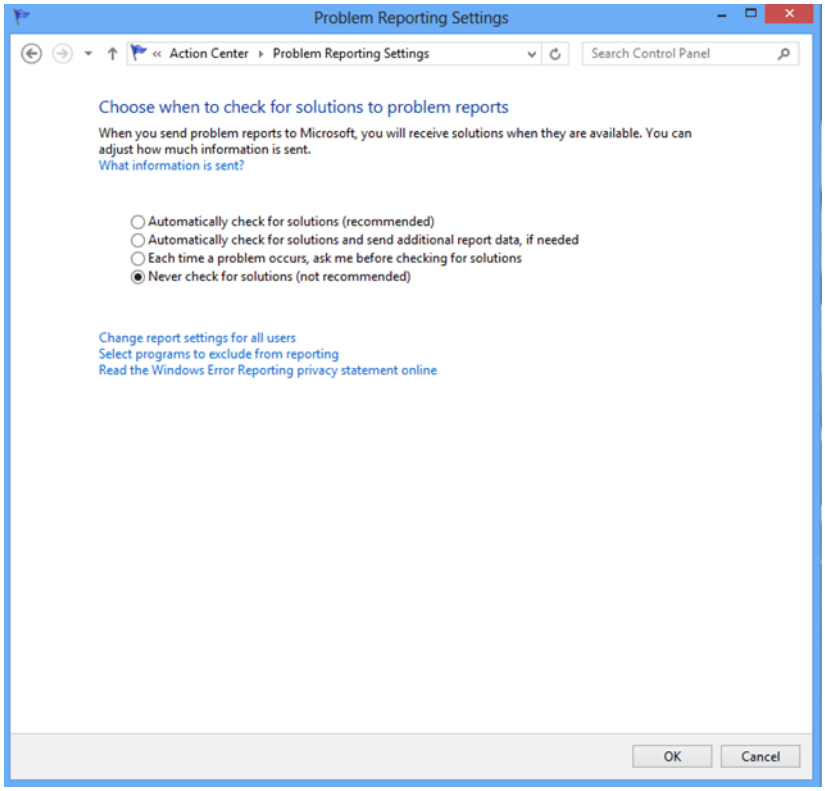
- From the desktop hold down the “Windows” key and type “I” to bring up the “Settings” charm, select “Control Panel”.
- Open the Action Center
- Select “Change Action Center Settings”:



- Select “Problem Reporting Settings”:



- Select “Never Check for Solutions”:



- Select “OK” twice and then close Action Center.

Addendum 2

Apply or Modify Auditing Policy Settings for a Local File or Folder

This was taken directly from Microsoft ~ https://technet.microsoft.com/en-us/library/Cc771070.aspx

Applies To: Windows 7, Windows Server 2008 R2

You can apply audit policies to individual files and folders on your computer by setting the permission type to record successful access attempts or failed access attempts in the security log.

Local Administrators is the minimum group membership required to complete this procedure. Review the details in "Additional considerations" in this topic.

To apply or modify auditing policy settings for a local file or folder

1. Open Windows Explorer.
2. Right-click the file or folder that you want to audit, click Properties, and then click the Security tab.
3. Click Edit, and then click Advanced.

*Note ~ if you are not logged on as a member of the Administrators group on this computer, you must provide administrative credentials to proceed.

4. In the Advanced Security Settings for <object> dialog box, click the Auditing tab.
5. Do one of the following:
    a. To set up auditing for a new user or group, click Add. In Enter the object name to select, type the name of the user or group that you want, and then click OK.
    b. To remove auditing for an existing group or user, click the group or user name, click Remove, click OK, and then skip the rest of this procedure.
    c. To view or change auditing for an existing group or user, click its name, and then click Edit.
6. In the Apply onto box, click the location where you want auditing to take place.
7. In the Access box, indicate what actions you want to audit by selecting the appropriate check boxes:
    a. To audit successful events, select the Successful check box.
    b. To stop auditing successful events, clear the Successful check box.
    c. To audit unsuccessful events, select the Failed check box.
    d. To stop auditing unsuccessful events, clear the Failed check box.
    e. To stop auditing all events, click Clear All.
8. If you want to prevent subsequent files and subfolders of the original object from inheriting these audit entries, select the Apply these auditing entries to objects and/or containers within this container only check box.

*Important ~ before setting up auditing for files and folders, you must enable object access auditing by defining auditing policy settings for the object access event category. If you do not enable object access auditing, you will receive an error message when you set up auditing for files and folders, and no files or folders will be audited.

Additional considerations

- You must be logged on as a member of the Administrators group or you must have been granted the Manage auditing and security log right in Group Policy to perform this procedure.
- To open Windows Explorer, click Start, point to All Programs, click Accessories, and then click Windows Explorer.
- After object access auditing is enabled, view the security log in Event Viewer to review the results of your changes.
- You can set up file and folder auditing only on NTFS drives.
- If you see either of the following, auditing has been inherited from the parent folder:
    - In the Auditing Entry for <File or Folder> dialog box, in the Access box, the check boxes are unavailable.
    - In the Advanced Security Settings for <File or Folder> dialog box, the Remove button is unavailable.
- Because the security log is limited in size, select the files and folders to be audited carefully. Also, consider the amount of disk space that you want to devote to the security log. The maximum size for the security log is defined in Event Viewer.

Addendum 3 ~ Coalfire's letter regarding CRL & P2PE

Terry Stevenson
NCR Connected Payments
85 Argonaut, ste 150
Aliso Viejo, CA 92656

December 21, 2015

Dear Terry,

Coalfire has reviewed the operations of NCR's Connected Payments P2PE solution for secure encryption of transaction at the pin pad located merchants' facilities. After assessing the design of the controls that are in place, it is Coalfire's opinion that the measures and practices listed below represent compensating controls that would allow Connected Payments to maintain Payment Card Industry (PCI) Payment Application Data Security Standard (PA-DSS) and DSS compliance for 2016, if it continues to implement these controls.

Our observations were that transactions cannot be decrypted until they arrive at NCR's secure decryption environment. The process that NCR's P2PE solution follows is:

- When the customer swipes, taps, or inserts their credit/debit card in the pin pad, the credit/debit card transactional message is encrypted with 168 bit 3DES DUKPT immediately upon capture.
- Each transaction is encrypted using a unique key derived by DUKPT (Derived Unique Key per Transaction).
- This encrypted message is then sent to the Point of Sale (POS) device, where it is encapsulated with TLS 1.2 and sent to Connected Payments' Cloud.
- In Connected Payments' Cloud, the transactional message is decrypted and sent to the acquiring banks' router, which is also located in Connected Payments' Cloud.
- Neither the POS, nor any part of the merchant environment has any access to the encryption keys.
- All encryption takes place in the pin pad and all decryption takes place in NCR's HSM located in NCR Connected Payments' Cloud.

NCR has become aware of issues with the TLS process for legacy POS infrastructure and has worked with Coalfire, its QSA, to clarify the TLS process as it pertains to certification validation process. NCR is required to have TLS1.2, strong ciphers, and strong certificate validation for its certifications: PCI PA-DSS 3.1 and PCI DSS 3.1. In keeping with these standards, NCR must require that all customers utilize TLS 1.2 and the strong ciphers when accessing NCR's Cloud from the Internet, even though the 3DES DUKPT encryption is a possible compensating control when dealing with weak TLS issues.

With regards to customers that are using legacy POS and are experiencing issues with latency during NCR's certificate validation process, customers can turn off certificate revocation and use the P2PE solution as a compensating control, since the strong

encryption when utilizing 3DES DUKPT at the pin pad satisfies compliance requirements for PCI DSS.

This position represents the opinion of Coalfire. This P2PE 3DES DUKPT encryption process must be evaluated as a compensating control with your QSA to ensure your QSA agrees with this opinion prior to disabling CRL checking. Further, this is not applicable for any clients running any version of NCR's software encryption.

Thank you.

Eric Hodge
QSA, CISA, CISSP
Managing Director, Southern California
Coalfire Systems
Eric.Hodge@Coalfire.com