Connected Payments

OpenEPS 829.*

PA-DSS 3.1 Implementation Guide

Version 3.0

July 2017

Confidential Information

Address correspondence to:

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

The product described in this book is a licensed product of NCR Corporation.

It is the policy of NCR Corporation (NCR) to improve products as new technology, components, software, and firmware become available. NCR, therefore, reserves the right to change specifications without prior notice.

All features, functions, and operations described herein may not be marketed by NCR in all parts of the world. In some instances, photographs are of equipment prototypes. Therefore, before using this document, consult with your NCR representative or NCR office for information that is applicable and current.

To maintain the quality of our publications, we need your comments on the accuracy, clarity, organization, and value of this book.

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
07/17/2017	David Means	Updated for 829.*, PA-DSS 3.1	3.0

This implementation guide is reviewed and updated as necessary, at least annually, 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.

Notice: Usage of Acronyms P2P and P2PE

OpenEPS IS NOT A PCI *VALIDATED P2PE SOLUTION* AND HAS NOT BEEN "ASSESSED IN ACCORDANCE WITH PCIS' P2PE STANDARD" NOR HAS IT BEEN "INCLUDED ON PCIS' LIST OF VALIDATED P2PE SOLUTIONS."

THE USE OF THE ACRONYM ***P2P*** OR ***P2PE*** IN THIS IMPLEMENTATION GUIDE IS USED IN MANNER CONSISTENT SOLELY WITH ITS HISTORICAL MEANING OF "*POINT TO POINT*" (P2P) or "*POINT TO POINT ENCRYPTION" (P2PE),* AND SHALL NOT BE CONFUSED TO MEAN OR INFER A *VALIDATED P2PE SOLUTION*.

DEPLOYMENT OF OpenEPS IN A CARDHOLDER ENVIRONMENT DOES NOT IN ANY WAY IMPLY, SUGGEST, OR AFFIRM THAT A PCI *VALIDATED P2PE SOLUTION* HAS BEEN DEPLOYED OR IMPLEMENTED IN YOUR CARDHOLDER DATA ENVIRONMENT.

For further information on PCI Validated P2PE Solutions, see here:

- https://www.pcisecuritystandards.org/pdfs/15_06_30%20PCI%20P2PE_v2_Press-Release.pdf

For further information on the definition of P2PE, as used in this PA-DSS Implementation Guide for OpenEPS 829.*, see here:

- Addendum

Table of Contents

Revision History ..... **iii**

Notice ..... **iv**

Notice: Usage of Acronyms P2P and P2PE ..... **v**

Table of Contents **6**

About this Document **9**

PA-DSS, PCI-DSS and Best Practices **10**

Executive Summary **11**

PCI Security Standards Council Reference Documents 11

Application Summary **13**

Data Flow Diagrams **19**

Generic Network Diagram 19
P2PE Hardware Encryption Data Flow 20
End to End Encryption Data Flow 22
Enhanced Software Encryption Data Flow 24
Manual PAN/CVV Entry Data Flow 26

Difference between PCI Compliance and PA-DSS Validation **28**

Requirements of PCI DSS 3.1 ..... **29**

Summary of PCI DSS v3.1 Requirements 29

Implementation of Payment Application in a PCI-Compliant Environment **30**

Remove Historical Sensitive Authentication Data (PA-DSS 1.1.4) 30
Handling of Sensitive Authentication Data (PA-DSS 1.1.5) 30
Secure Deletion of Cardholder Data (PA-DSS 2.1) 31
Secure Deletion of Pre-Authorization Data 31
Procedure for inadvertent capture of CHD 32
Procedure for Deleting Off-line Files 32

All PAN is Masked by Default (PA-DSS 2.2) 33
Cardholder Data Encryption & Key Management (PA-DSS 2.3, 2.4, and 2.5) 33
PAN/SAD Encryption 33
POS Provided PAN 33
Store-and-Forward: Internet Outage 34
Key Management 34

Removal of Historical Cryptographic Material (PA-DSS 2.6) 36

Set up Strong Access Controls (3.1 and 3.2) 37
Properly Train and Monitor Admin Personnel 42
Log settings must be compliant (PA-DSS 4.1.b, 4.4.b) 42
PCI-Compliant Wireless settings (PA-DSS 6.1.a and 6.2.b) 43
Services and Protocols (PA-DSS 8.2.c) 45
Never store cardholder data on internet-accessible systems (PA-DSS 9.1.c) 46
PCI-Compliant Remote Access (10.1) 46
PCI-Compliant Delivery of Updates (PA-DSS 10.2.1.a) 46
PCI-Compliant Remote Access (10.2.3.a) 47
Data Transport Encryption (PA-DSS 11.1.b) 48
PCI-Compliant Use of End User Messaging Technologies (PA-DSS 11.2.b) 49
Non-console administration (PA-DSS 12.1) 49
Network Segmentation 49
Maintain an Information Security Program 49
Application System Configuration 50
Payment Application Initial Setup & Configuration 52
Addendum – Installation Instructions 54
Acquiring the Connected Payments Software 54
Connected Payments Transaction Management Portal 55
OpenEPS Installation and Configuration 56
Installation Process 56
Test Connected Payments Connectivity (Optional) 64
Addendum – Addressing Inadvertent Capture of PAN 68
Addressing Inadvertent Capture of PAN on Windows 7 68
Disable System Restore Settings 68
Encrypt the System PageFile.sys 69
Clear the System Pagefile.sys on shutdown 71
Disable System Management of Pagefile.sys 72
Disable Windows Error Reporting 74
Addressing Inadvertent Capture of CHD on Windows 8.1 77
Disabling System Restore – Windows 8.1 77
Encrypting PageFile.sys – Windows 8.1 79
Clear the System Pagefile.sys on shutdown 81
Disabling System Management of PageFile.sys – Windows 8.1 82
Disabling Windows Error Reporting – Windows 8.1 85
Addressing Inadvertent Capture of Pan on Windows 10 89
Disabling System Restore – Windows 10 89
Encrypting PageFile.sys – Windows 10 91
Clear the System Pagefile.sys on shutdown 92
Disabling System Management of PageFile.sys – Windows 10 93
Disabling Windows Error Reporting – Windows 10 96
Addendum – File/Folder Auditing Policy Settings 99
Apply or Modify Auditing Policy Settings for a Local File or Folder 99
Addendum – Certificate Validation & Configuration 101

Certificate Validation 101
Certificate Validation & Off-line Processing 101
CRL Checking on public networks for P2PE 102
Certificate Pinning 105

Configurations for CRL/OCSP Processing 105
Discovery 105
Information Gathering Procedures 105
Example: Gathering Revocation Information 106

Firewall / Proxy Configuration Guidance 107
Example CRL/OCSP URI 108
Example Regular Expressions 108

Summary 108
Addendum – TLS and FIPS 140-2 109

FIPS 140-2 109
TLS v1.2 109
Addendum – Firewall Configuration 111

Keeping the Internet Out 111
Outbound and Inbound Connections 111
Firewalls 112
Additional Safety Measures 112

Connections: Trusted Software to Trusted Sites 113
POS Lane Connections 113
Firewall Configuration Information 114
Report Service, Web Site Access 114
Firewall Configuration Example 116

Addendum – Definitions 120

Addendum – Table of Figures 124

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 July 2015*).

PA-DSS, PCI-DSS and Best Practices

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, 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 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 (PA-QSA):

Coalfire Systems, Inc. 11000 Westmoor Circle Suite 450 Westminster, CO 80021	Coalfire Systems, Inc. 1633 Westlake Ave N #100 Seattle, WA 98109

This document also explains the Payment Card Industry initiative and the Payment Application Data Security Standard 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' PCI POI Device approved (point of interaction) 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 POI Device 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
&
Stored Sensitive Account Data</td><td colspan="3">Overview of Storage of Encrypted PAN/SAD pre-Authorization

Assumptions

a) Connected Payments is unreachable due to
i) Internet / Intranet Disruption or Outage
ii) Certificate validation failure
iii) Other hardware failure, local or remote
b) POI Device
i) POI device performs encryption of PAN/SAD
ii) OpenEPS does not have access to keys used by the POI device to encrypt PAN/SAD
iii) POI Device is out-of-scope for pre-authorization, stored PAN/SAD
c) POS Device Encryption
i) OpenEPS has access to PAN/SAD encryption key</td></tr>
</table>

	ii) OpenEPS performs two layers of encryption of PAN/SAD **Brief Description** a) OpenEPS monitors availability (online status) of Connected Payments (CP) b) When CP is down, OpenEPS saves encrypted transactions pre-authentication to disk. c) Encrypted transactions (from the POI device or from OpenEPS) are wrapped (re-encrypted) using AES-256 when being saved to disk. **Detailed Description** OpenEPS detects the unavailability of Connected Payments. OpenEPS then performs the following actions: 1) OpenEPS performs store and forward, as follows: a) The off-line data is encrypted as follows i) Encryption #1 - CHD is encrypted by POI Device using standard processing rules (1) See ***Application Encryption*** (below) for more information regarding how CHD is encrypted. ii) Encryption #2 - The encrypted PAN/SAD is additionally encrypted using AES 256 2) The double-encrypted transaction record is stored in the off-line transaction file. 3) When Connected Payments is determined to be available, the following actions are performed: a) OpenEPS retrieves a stored off-line data transaction and places it in RAM. b) OpenEPS removes the AES-256 encryption from the encrypted transaction in RAM. c) OpenEPS sends the encrypted PAN/SAD to Connected Payments. d) When a response is received from Connected Payments for a given transaction, the following is performed: i) The off-line data transaction is marked as processed. ii) The off-line data transaction is erased with null bytes. iii) While unprocessed records exist, continue process at ***Step 3*** 4) When all records have been processed, a) The off-line data file is deleted and then re-initialized for the next time Connected Payments is unavailable 5) If there is stored off-line data that does not forward a) A configurable threshold is updated and monitored b) When the threshold is reached, an alert is triggered indicating the transactions are not being forwarded to Connected Payments

<table>
<tr><td>Components of the Payment Application</td><td>The only component that is installed in a typical merchant environment is the OpenEPS DLL, as bundled with the POS Application.

No modifications are allowed to the OpenEPS DLL.

• Integrity checking is built into the application using industry standard digital signing and validation methods.

• OpenEPS performs an external referential check with Connected Payments to further ensure the application has not been modified.</td></tr>
<tr><td>Required Third Party Payment Application Software</td><td>No third party dependencies</td></tr>
<tr><td>POI Device Compatibility</td><td><table><tr><th>OpenEPS Version</th><th>PCI Approved POI Device</th></tr><tr><td>829.*</td><td>Equinox, Ingenico, VeriFone Vx (RBA)</td></tr><tr><td>Notes</td><td rowspan="2">Other devices may be supported that are not listed above. For the latest information regarding supported POI device for your version of OpenEPS, contact ConnectedSupport@retalix.com.</td></tr><tr><td></td></tr></table></td></tr>
<tr><td>Database Software Supported</td><td>OpenEPS does not have a database and it does not support traditional database software to store application data. OpenEPS utilizes an internal proprietary flat file system to store data.</td></tr>
<tr><td>Operating System(s) Supported & Validated</td><td>OpenEPS has been PA-DSS v3.1 validated with the following versions of Windows:

• POSReady 2009, SP3
• POSReady 7 SP1
• Windows 7 SP1 Enterprise
• Windows 8.1 Enterprise
• Windows 10 Enterprise</td></tr>
<tr><td>Application Authentication</td><td>Authentication

• Between the POI Device and OpenEPS
○ OpenEPS authentication between a POI Device and OpenEPS is through a session key.
• Between OpenEPS and the POS
○ None
• Between OpenEPS and ServerEPS / Connected Payments
○ TLS 1.2 authentication with ServerEPS using Certificate validation</td></tr>
</table>

	○ Certificate Pinning on the Subject Information ○ Company number and store number validated by ServerEPS
Application Encryption Summary	**Point to Point Hardware Encryption** • OpenEPS leverages a POI Device approved hardware encryption solution employing 3DES 168 bit DUKPT, using 168-bits. **End to End Hardware Encryption** • OpenEPS may Leverage a POI Device approved hardware encryption solution employing an encryption scheme at the discretion of the Payer. **Enhanced Software Encryption** This information is retained for historical purposes only. NCR Connected Payments is not currently deployed using Enhanced Software Encryption (ESE). • For merchants who have not upgraded their POI Device and are still using ESE, the following rules apply: ○ OpenEPS provides a SALT to the POI Device and an RSA 4096 KEK ○ The POI Device generates a DEK using AES 128 ○ The POI Device encrypts the PAN/SAD using DEK ○ The POI Device encrypts the DEK using the KEK **POS Manual Entry of PAN/SAD** For POS Systems that allow PAN/SAD input by the cashier: • OpenEPS uses dynamically generated keys to encrypted PAN/SAD using AES-128 • Keys are dynamically derived at the time of the transaction • Keys cannot be managed by the user • For more information, see: Cardholder Data Encryption & Key Management **Off-line Transactions** • Off-line data files are used to store transactions when connectivity to Connected Payments is interrupted, as follows: ○ OpenEPS receives encrypted PAN/SAD ($(M)_e$) from POI Device ○ OpenEPS encrypts the $(M)_e$ with AES-256 resulting in $E((M)_e)$ ○ OpenEPS stores $E((M)_e)$ in the off-line data file. ▪ Information is stored until communication with Connected Payments can be resumed. For more information, see the following sections: • Secure Deletion of Cardholder Data • Cardholder Data Encryption & Key Management

<table>
<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 Device</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 POI Device a payment request. The customer conducts either a card swipe, EMV or NFC. The POI Device encrypts the transactional data using either 3DES, 168 bit DUKPT or Enhance software AES 128 bit depending upon the POI Device utilized. POI Device sends this encrypted information in the form of a blob to OpenEPS.

OpenEPS encapsulates the encrypted blob with TLS v1.2 session encryption (see Addendum - TLS and FIPS 140-2) and sends the information to Connected Payments for processing. Connected Payments decrypts the session encryption, decrypts the transaction (3DES blob or AES blob), formats the transaction to the payment processor’s requirements and forwards the transaction for authorization to the appropriate service provider.

Once Connected Payments receives the authorization message from the Payer, then ServerEPS sends back to OpenEPS the response of approval/decline. OpenEPS then relays the information to the POI Device & POS.</td></tr>
<tr><td>Description of Listing Versioning Methodology</td><td>There are three main OpenEPS modules used. The OpenEPS module is named “MTX_EPS.dll”

The OpenEPS modules are versioned based on a 4-level versioning methodology numbering scheme as follows:

Stream . Version . Flag . Build Number

The Stream number is updated when significant and PA-DSS impactful development changes are made.

The Version number is updated when development or PCI impactful updates are made.

The Flag number indicates the capabilities of the mtx_eps.dll module. It will be 0 in all other modules and in unreleased versions of mtx_eps.dll. This number does not affect PA-DSS requirements, neither does it affect security in general. It is designated with the wildcard which is represented by the single “*” (asterisk) character.</td></tr>
</table>

The ***Build*** number designates the specific build number of the software, which is an NCR internal tracking number. This number does not affect PA-DSS requirements, neither does it affect security in general. It is designated with the wildcard which is represented by the single "*" (asterisk) character.

On the PCI website, ***Flag*** and ***Build*** never impact PA-DSS Requirements or application security. Only one wildcard character is included in the listed version number to represent the ***Flag*** and ***Build*** numbers.

Software	Example	Stream	Version	Flags	Build
MTX_EPS.dll	828.21.12	828	1	21	12
MTX_SE.dll	829.1.0.2	829	1	0	2
MTX_POS.dll	829.1.0.1	829	1	0	1

Data Flow Diagrams

Generic Network Diagram

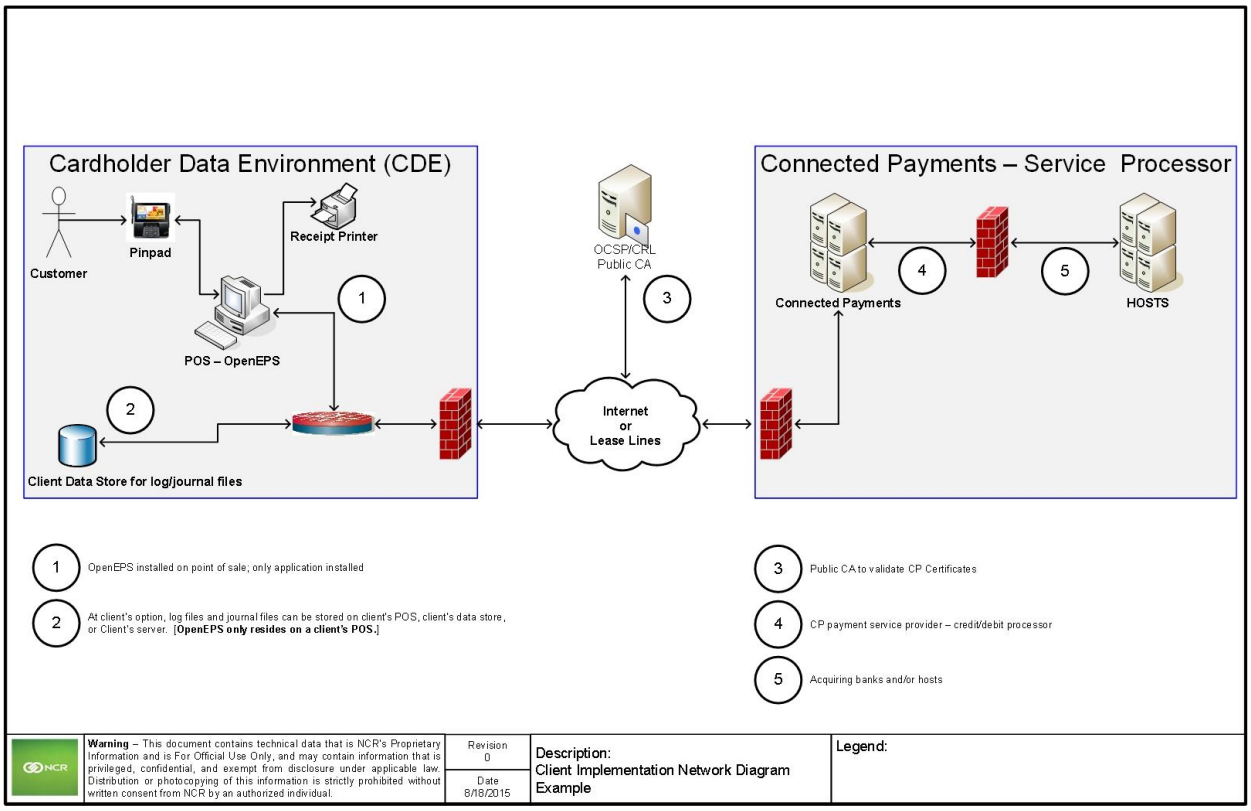
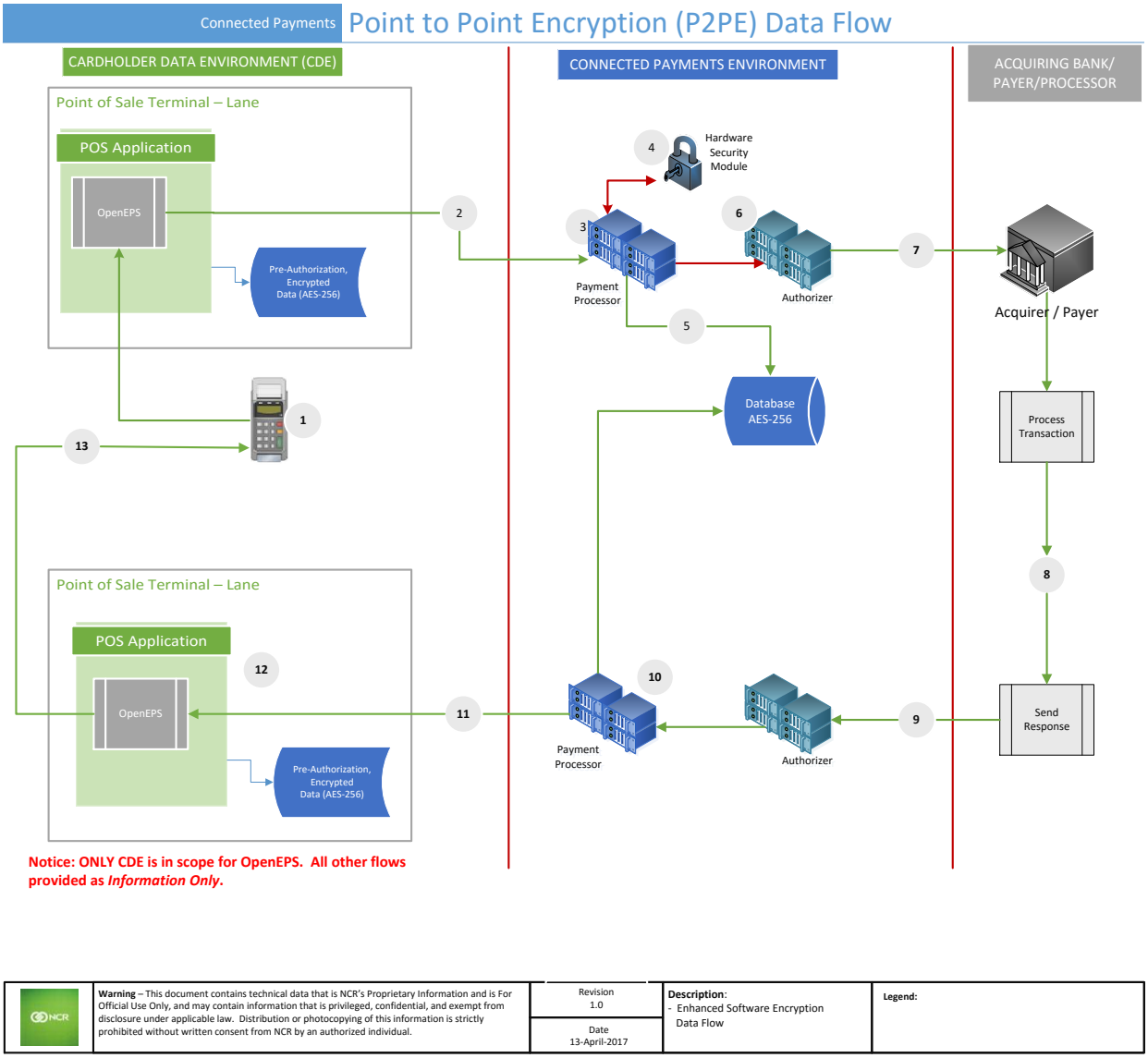


Figure 1 – Generic Network Dataflow Diagram

P2PE Hardware Encryption Data Flow



NCR	**Warning** – This document contains technical data that is NCR's Proprietary Information and is For Official Use Only, and may contain information that is privileged, confidential, and exempt from disclosure under applicable law. Distribution or photocopying of this information is strictly prohibited without written consent from NCR by an authorized individual.	Revision 1.0 Date 13-April-2017	**Description**: - Enhanced Software Encryption Data Flow	**Legend:**

Figure 2 - P2P Data Flow Diagram

Note: Depending upon POS Application & Business Processing rules, the Manual PAN/CVV Data Flow may be an optional data flow for this diagram.

1. The customer uses their card at the POI Device. Data is encrypted using 3DES (TDEA) DUKPT
2. OpenEPS establishes a TLS v1.2 session to Connected Payments (see Addendum - TLS and FIPS 140-2 for more information).
   2.1. OpenEPS establishes an HTTPS connection to Connected Payments using TLS v1.2
   2.2. OpenEPS sends encrypted PAN/SAD to Connected Payments (CP)
3. Connected Payments (CP) accepts the encrypted session (HTTPS)

3.1. Processor begins Transaction State Management tasks
4. Processor sends the encrypted PAN/SAD to the Hardware Security Module (HSM) for decryption
   4.1. The HSM decrypts the PAN/SAD
   4.2. The HSM encrypts portions of the PAN/SAD for temporary, pre-authentication storage
5. Processor stores data in the transaction table
   5.1. Stores encrypted (HSM) data as necessary for processing
   5.2. Stores other transaction information as necessary for the Merchant Environment (e.g., Retail, Hospitality, etc.)
6. Processor sends the un-encrypted PAN/SAD to the Authorizer
7. The Authorizer handles the PAN/SAD according to Payer Rules, as follows
   7.1. Data is transmitted to Payer using
      7.1.1. Payer's routers
      7.1.2. Payer's leased Lines
      7.1.3. Payer's Encryption rules
8. Payer processes transaction
   8.1. Payer generates approval (approved/declined)
9. Payer sends approval response with transaction ID
   9.1. No PAN/SAD is in the response.
10. Authorizer forwards response to Processor
   10.1. Processor updates transaction database with payer transaction ID
11. Processor responds to OpenEPS (using session established in *Step 3)* with approval
   11.1. No PAN/SAD is included in the response
12. OpenEPS updates POS Device with Approval Message
   12.1. OpenEPS Securely deletes encrypted, Pre-Authorization data.
   12.2. See Secure Deletion of Cardholder Data for more information.
13. OpenEPS updates POI Device with Approval Message

End to End Encryption Data Flow

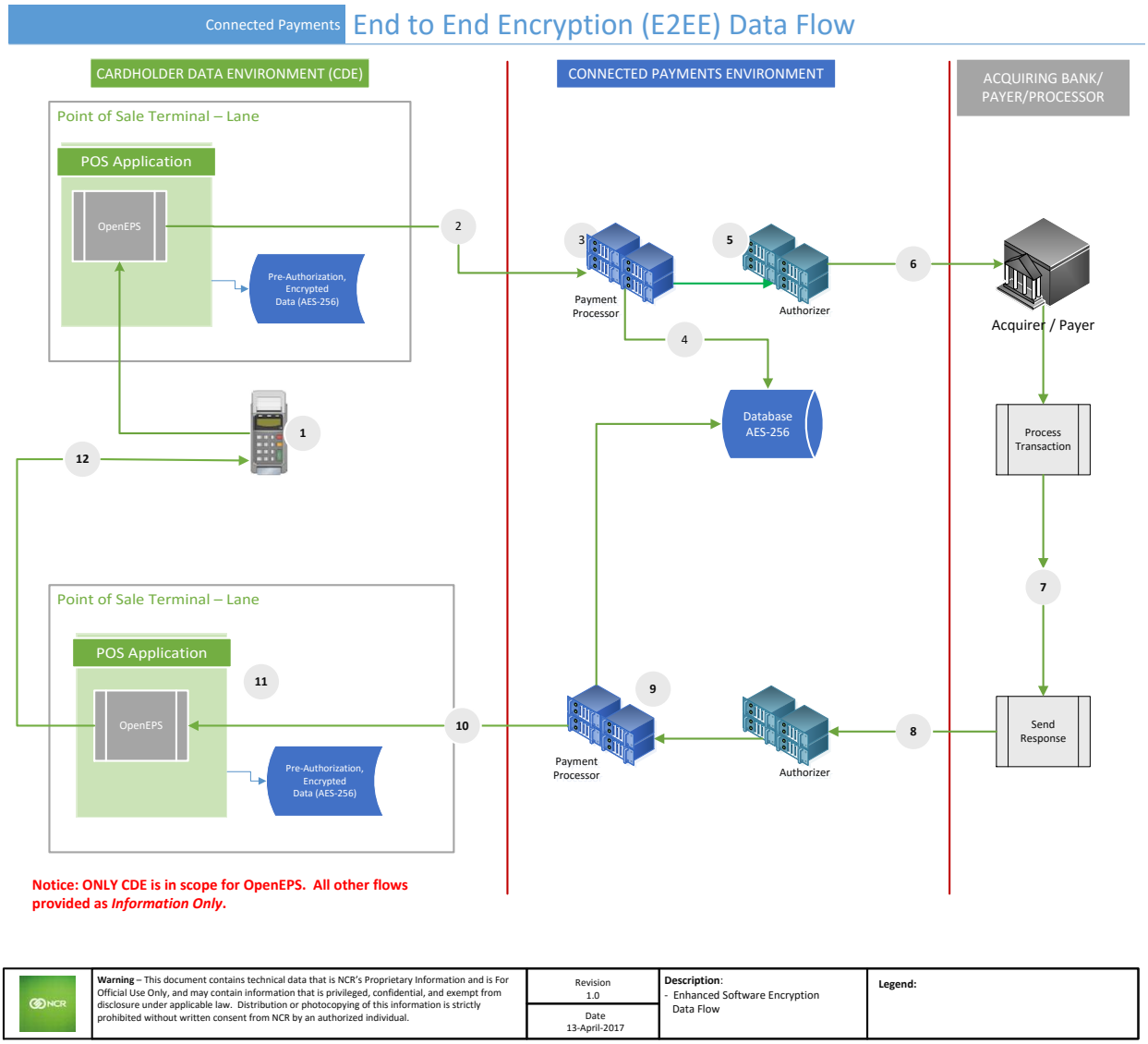


Figure 3 - E2EE Data Flow Diagram

Note: Depending upon POS Application & Business Processing rules, the Manual PAN/CVV Data Flow may be an optional data flow for this diagram.

1. The customer uses their card at the POI Device.
   1.1. Data is encrypted using Payer Encryption Scheme
2. OpenEPS establishes a TLS v1.2 session to Connected Payments (see Addendum - TLS and FIPS 140-2 for more information).
   2.1. OpenEPS establishes an HTTPS connection to Connected Payments using TLS v1.2
   2.2. OpenEPS sends encrypted PAN/SAD to Connected Payments (CP)

3. Connected Payments (CP) accepts the encrypted session (HTTPS)
   3.1. Processor begins Transaction State Management tasks
4. Processor stores updates the transaction table
   4.1. Stores other transaction information as necessary for the Merchant Environment (e.g., Retail, Hospitality, etc.)
5. Processor sends the POI Device Encrypted Transaction to the Authorizer
6. The Authorizer sends the POI Device Encrypted Transaction to the Payer
7. Payer processes transaction
   7.1. Decrypts the POI Device Transaction
   7.2. Payer generates approval (approved/declined)
8. Payer sends approval response with transaction ID
   8.1. No PAN/SAD is in the response.
9. Authorizer forwards response to Processor
   9.1. Processor updates transaction database with payer transaction ID
10. Processor responds to OpenEPS (using session established in *Step 3)* with approval
    10.1. No PAN/SAD is included in the response
11. OpenEPS updates POS with Approval Message
    11.1. OpenEPS Securely deletes encrypted, Pre-Authorization data.
    11.2. See Secure Deletion of Cardholder Data for more information.
12. OpenEPS updates POI Device with Approval Message

Enhanced Software Encryption Data Flow

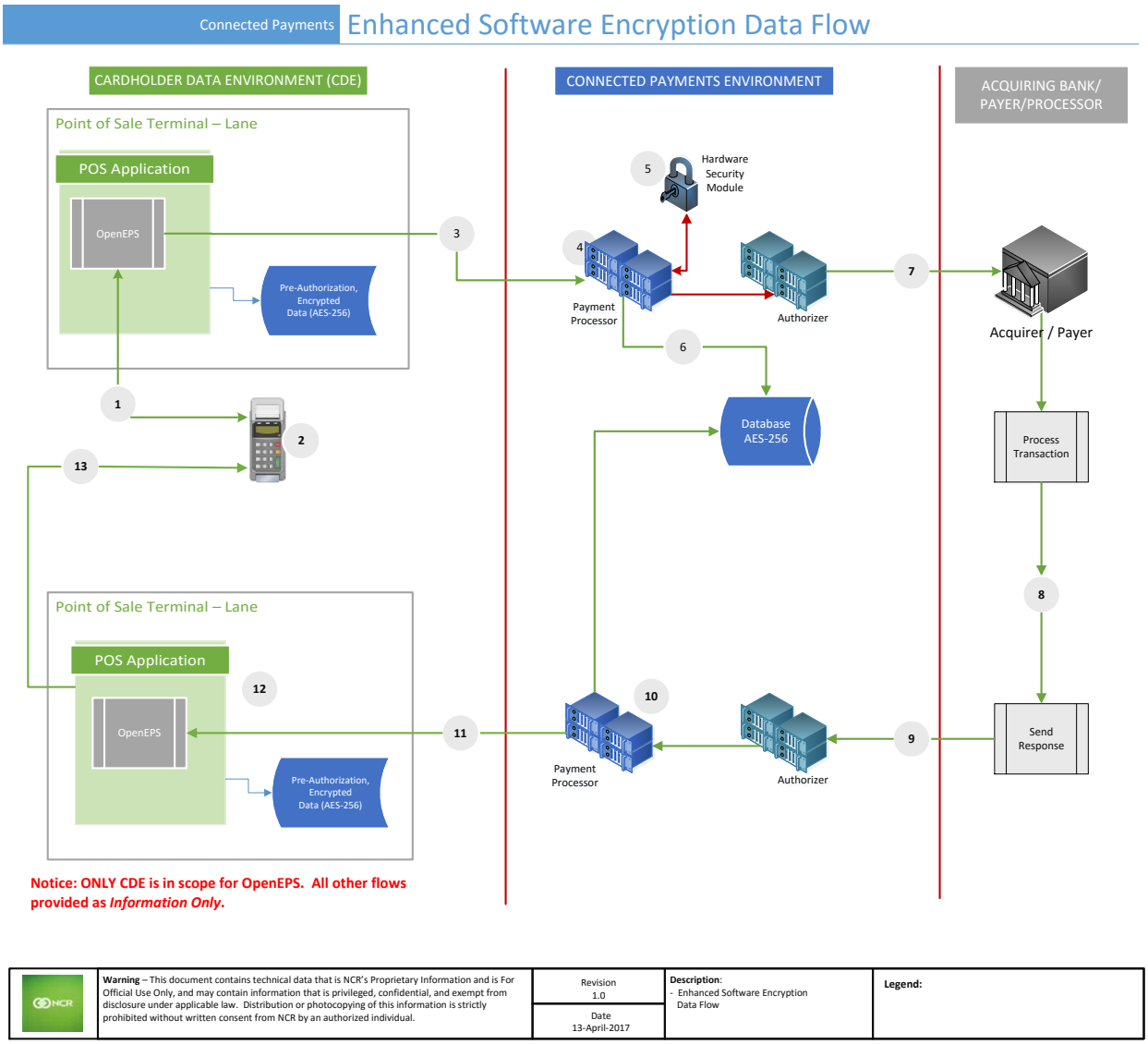


Figure 4 – ESE Data Flow Diagram

Note: Depending upon POS Application & Business Processing rules, the Manual PAN/CVV Data Flow may be an optional data flow for this diagram.

1. Merchant logon to POS
   1.1. OpenEPS creates SALT for AES 128 key
   1.2. OpenEPS retrieves RSA 4096 KEK from Connected Payments
   1.3. OpenEPS provides SALT and KEK to POI Device
   1.4. POI Device Creates AES 128 DEK
2. The customer uses their card at the POI Device.

2.1. Data is encrypted using DEK
2.2. DEK is encrypted using KEK
3. OpenEPS establishes a TLS v1.2 session to Connected Payments (see Addendum - TLS and FIPS 140-2 for more information).
3.1. OpenEPS establishes an HTTPS connection to Connected Payments using TLS v1.2
3.2. OpenEPS sends encrypted PAN/SAD to Connected Payments (CP)
4. Connected Payments (CP) accepts the encrypted session (HTTPS)
4.1. Processor begins Transaction State Management tasks
5. Processor sends the encrypted PAN/SAD to the Hardware Security Module (HSM) for decryption
5.1. The HSM decrypts the PAN/SAD
5.2. The HSM encrypts portions of the PAN/SAD for temporary, pre-authentication storage
6. Processor stores updates the transaction table
6.1. Stores encrypted (HSM) data as necessary for processing
6.2. Stores other transaction information as necessary for the Merchant Environment (e.g., Retail, Hospitality, etc.)
6.3. Processor sends the un-encrypted PAN/SAD to the Authorizer
7. The Authorizer handles the PAN/SAD according to Payer Rules, as follows
7.1. Data is transmitted to Payer using
7.1.1. Payer's routers
7.1.2. Payer's leased Lines
7.1.3. Payer's Encryption rules
8. Payer processes transaction
8.1. Payer generates approval (approved/declined)
9. Payer sends approval response with transaction ID
9.1. No PAN/SAD is in the response.
10. Authorizer forwards response to Processor
10.1. Processor updates transaction database with payer transaction ID
11. Processor responds to OpenEPS (using session established in *Step 3)* with approval
11.1. No PAN/SAD is included in the response
12. OpenEPS updates POS with Approval Message
12.1. OpenEPS Securely deletes encrypted, Pre-Authorization data.
12.2. See Secure Deletion of Cardholder Data for more information.
13. OpenEPS updates POI Device with Approval Message

Manual PAN/CVV Entry Data Flow

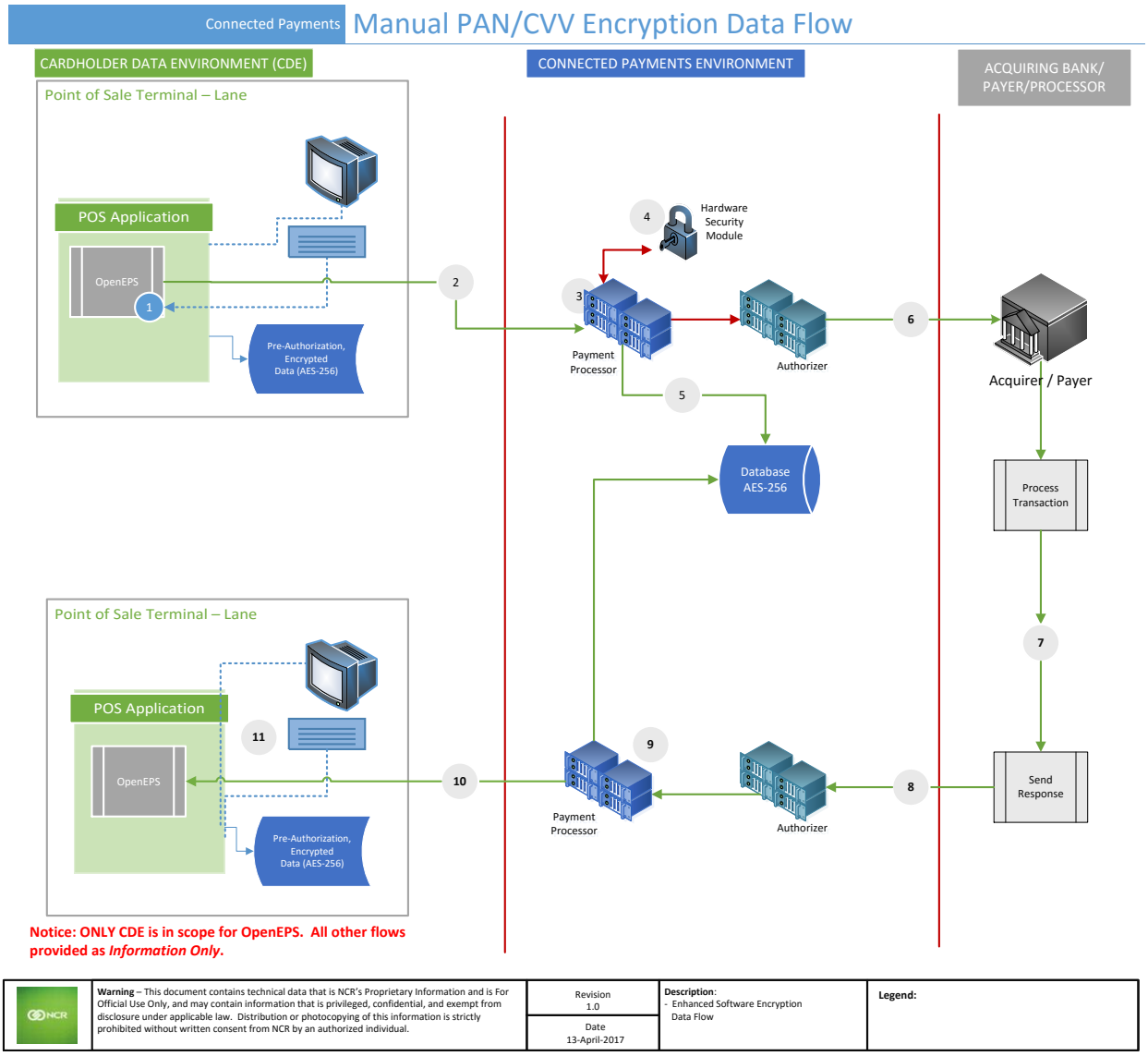


Figure 5 - Manual PAN/CVV Data Entry Flow

Note: New (non-legacy) POS systems generally do not allow manual entry of PAN/CVV data. Nevertheless, manual entry of PAN/CVV data is entirely controlled by the POS System – it is not an OpenEPS controlled function.

As such, this flow may be valid for any deployment (P2PE, E2EE and ESE), depending upon the Business Rules designed into your POS System. For more information and configuration details, please see the POS User and Implementation Guide provided with your POS System.

1. Merchant logon to POS System
   1.1. Merchant manually enters PAN/CVV data

1.2. OpenEPS creates a dynamic encryption key using AES-128
1.2.1. For more information regarding dynamic key generation, see Cardholder Data Encryption & Key Management
2. OpenEPS establishes a TLS v1.2 session to Connected Payments (see Addendum - TLS and FIPS 140-2 for more information).
2.1. OpenEPS establishes an HTTPS connection to Connected Payments using TLS v1.2
2.2. OpenEPS sends encrypted PAN/SAD to Connected Payments (CP)
3. Connected Payments (CP) accepts the encrypted session (HTTPS)
3.1. Processor begins Transaction State Management tasks
4. Processor sends the encrypted PAN/SAD to the Hardware Security Module (HSM) for decryption
4.1. The HSM decrypts the PAN/SAD
4.2. The HSM encrypts portions of the PAN/SAD for temporary, pre-authentication storage
5. Processor stores updates the transaction table
5.1. Stores encrypted (HSM) data as necessary for processing
5.2. Stores other transaction information as necessary for the Merchant Environment (e.g., Retail, Hospitality, etc.)
5.3. Processor sends the un-encrypted PAN/SAD to the Authorizer
6. The Authorizer handles the PAN/SAD according to Payer Rules, as follows
6.1. Data is transmitted to Payer using
6.1.1. Payer's routers
6.1.2. Payer's leased Lines
6.1.3. Payer's Encryption rules
7. Payer processes transaction
7.1. Payer generates approval (approved/declined)
8. Payer sends approval response with transaction ID
8.1. No PAN/SAD is in the response.
9. Authorizer forwards response to Processor
9.1. Processor updates transaction database with payer transaction ID
10. Processor responds to OpenEPS (using session established in *Step 3)* with approval
10.1. No PAN/SAD is included in the response
11. OpenEPS updates POS with Approval Message
11.1. OpenEPS Securely deletes encrypted, Pre-Authorization data.
11.2. See Secure Deletion of Cardholder Data for more information.

Difference between PCI Compliance and PA-DSS Validation

As a software vendor who develops payment applications, it is our responsibility 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 3.1

PCI DSS is a requirement for all cardholder data environments. Since OpenEPS is not a *PCI P2PE Validated Solution,* you must observe, implement and employ the services of a QSA to audit your cardholder environment and the PCI DSS approved controls you have deployed in your environment. PCI DSS consists of 12 domains, as follows below.

For a prioritized implementation approach for PCI DSS v3.1, refer to the "Prioritized Approach for PCI DSS v3.1," here: https://www.pcisecuritystandards.org/documents/Prioritized-Approach-for-PCI_DSS-v3_2.pdf.

Otherwise, you may obtain a copy PCI DSS v3.1 standards (any many other PCI related documents) from the PCI SSC Document Library, here: https://www.pcisecuritystandards.org/document_library

Summary of PCI DSS v3.1 Requirements

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
3. Protect stored cardholder data
4. Encrypt transmission of cardholder data across open, public networks
5. Use and regularly update anti-virus software or programs
6. Develop and maintain secure systems and applications
7. Restrict access to cardholder data by business need to know
8. Assign a unique ID to each person with computer access
9. Restrict physical access to cardholder data
10. Track and monitor all access to network resources and cardholder data
11. Regularly test security systems and processes
12. 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 advice of our QSA, the guidance below does not include store and forward scenarios.

Remove Historical Sensitive Authentication Data (PA-DSS 1.1.4)

Sensitive Authentication Data (SAD) includes security-related information, PINs, and PIN blocks and PAN data used to authenticate cardholders and/or authorize payment card transactions.

Previous versions of OpenEPS were not designed to permit the retention of historical, sensitive authentication data (CHD, PAN or SAD). 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, and we strongly recommend that you do not do this either. However, if for any reason you should do so, the following guidelines must be followed when dealing with Sensitive Authentication Data used for pre-authorization (swipe data, validation values or codes, PIN or PIN block data):

- Collect sensitive authentication data only when needed to solve a specific problem
- Store such data only in specific, known locations with limited access
- Collect only the limited amount of data needed to solve a specific problem
- Encrypt sensitive authentication data while stored
- Securely delete such data immediately after use

Secure Deletion of Cardholder Data (PA-DSS 2.1)

OpenEPS routinely stores encrypted transaction data only for the duration of the transaction. Once the transaction is complete, the data is securely deleted using methodologies compliant Department of Defense standard DOD 5220.22-M. Therefore, during the normal operations OpenEPS, you are never required to maintain or routinely delete encrypted, and stored CHD: OpenEPS automatically performs secure deletion of encrypted CHD automatically.

When decommissioning a POS system, you should follow these instructions to insure that any remnants of encrypted, CHD has been thoroughly removed.

OpenEPS may store encrypted transaction to disk data by wrapping the encrypted blob an additional encryption layer using AES 256. The double-encrypted blob will be stored to the file system in one or more the following files, depending upon the operating mode of the POS:

Off-Line / Transaction Files	
File Name	**Description**
`off*.eft`	Encrypted: Off-line transactions, signature/receipts, declined advice transaction
`tor*.eft`	Encrypted: Time-out Reversals
`towineps*.eft`	Encrypted: Temporary, pre-authorization, transaction data
`PriorTrans*.eft`	No CHD: used to allow voiding of prior transactions when off-line

Note: replace the asterisk (*) with the lane numbers in order to reveal the actual file name. In example, *Lane 1*, would result in `off0001.eft`, `tor0001.eft`, etc.

Secure Deletion of Pre-Authorization Data

Pre-Authorization data is temporarily stored. This data is encrypted by the POI Device, and OpenEPS does not have access to PAN/SAD. When storing data encrypted by OpenEPS (see Manual PAN/CVV Entry Data Flow) OpenEPS first encrypts the data using AES-128, then re-encrypts the data using AES-256 when storing it to disk.

All transaction data is encrypted using AES-256 and temporarily stored to disk. Once the transaction is complete, the file is securely deleted using the method referenced above, Secure Deletion of Cardholder Data (PA-DSS 2.1).

Procedure for inadvertent capture of CHD

1) To protect against inadvertent storage of CHD, see Addendum - Addressing Inadvertent Capture of PAN.

Procedure for Deleting Off-line Files

Warning: Performing this procedure during normal operations (while transactions are pending or processing) will result in the loss of one or more transactions.

1) Obtain a secure file delete tool, such as SysInternals' "`sdelete`"
    a. See: https://technet.microsoft.com/en-us/sysinternals/sdelete.aspx
2) Ensure that all pending transactions have been processed
3) Using Windows Explorer, browse to the OpenEPS Directory in order to visualize the directory and its contents:
    - `C:\Program Files\MicroTrax\OpenEPS`

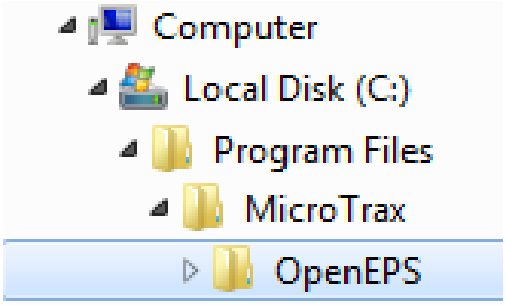


Figure 6 - OpenEPS Directory

4) Observe for **`*.eft`** files (as defined above) in this directory.
5) Use the "cmd" or "powershell" terminal command and navigate to the directory shown above.
6) Use "sdelete" (or similar tool) to securely delete the off-line files (as documented above), as follows:
    - `cd "C:\Program Files\MicroTrax\OpenEPS"`
    - `sdelete -p 10 [off-line-file].eft`

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

The POI Device may provide the first 6, last 4 of the PAN to OpenEPS. However, OpenEPS does not have the ability to display full PAN for any reason. There are no user controlled configurations available and therefore there are no configuration details to be provided as required for PA-DSS v3.1.

Depending upon Host (payer) requirements, the POS may or may not make use of a receipt generated by OpenEPS. In this event, PAN will be masked (as derived from the first 6 and last 4 digits of the PAN), by displaying only the last 4 digits of the PAN.

The POS may request a masked PAN from OpenEPS. The masked PAN will be returned as the first 6 digits of the card number, along with the last 4 digits, which are the maximum displayable characters allowed by PCI compliance. Once the masked PAN is received by the POS, it will determine how to present the masked PAN using pre-defined business rules.

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

PAN/SAD Encryption

In order to guard against inadvertent capture of PAN, refer to Addendum - Addressing Inadvertent Capture of PAN. All PAN/SAD must be rendered unreadable anywhere it is stored (including data on portable digital media, backup media, and in logs).

POS Provided PAN

In the event the POS application accepts manual entry of PAN/SAD, then OpenEPS uses an encryption methodology with dynamically generated keys to automatically encrypt PAN/SAD. These keys are dynamically derived at the time of transaction and cannot be managed by the user. Keys are automatically cleared from memory after the transaction is completed and are never stored to disk.

Store-and-Forward: Internet Outage

When it is determined that Connected Payments is off-line, then the encrypted blob (as described above, or as received from the POI device) is forwarded to the store-and-forward module of OpenEPS. This module adds an additional layer of AES-256 encryption when storing it to disk.

An encryption methodology with dynamically generated keys is utilized in the store-and-forward module. Once the encryption process is complete, the memory structures used by OpenEPS are immediately cleared so that no PAN/SAD is retained in the heap or stack structures used by OpenEPS.

- OpenEPS does not output PAN/SAD for use or storage in a merchants' environment for any reason therefore there are no location or configuration details to provide as required by PA-DSS v3.1.
- OpenEPS does not have a debugging mode that could write PAN to debugging logs.
- OpenEPS does not output PAN/SAD to journals or system log/event files

OpenEPS uses a dynamic key encryption methodology and there are no key management activities required by customers or resellers/integrators. The following key management functions are performed automatically using AES 256 dynamic encryption key methodology and there are no key custodians or intervention required by customers or resellers/integrators for the following activities:

- Generation of strong cryptographic keys.
- Secure cryptographic key distribution
- Secure cryptographic key storage
- Cryptographic key changes for keys that have reached the end of their cryptoperiod.
- Separation of duties (key material is split between OpenEPS and ServerEPS).

Manually replace keys when it is suspected that the integrity of key material has been weakened and/or when known or suspected compromise is detected. If retired or replaced cryptographic keys are retained, the application cannot use these keys for encryption operations. See *Key Management*, below for instructions.

Key Management

OpenEPS uses dynamic key encryption methodology in the following scenarios:

- To Encrypted PAN as entered by the POS
- To establish as session with the POI Device

- To wrap transactions from the POI Device or POS terminal when in the “off-Line” mode of operation.
    - “Off-Line” mode processing is independent of
        - Enhanced Software Encryption
        - P2P Encryption
        - E2E Encryption
        - POS Entry of PAN data

Dynamic key generation is used in all cases where needed:

1) For encryption of POS entered PAN. Dynamic encryption key methodology using AES 128 is implemented in OpenEPS.
2) For encryption encrypted PAN data while in “off-line” mode. Dynamic encryption key methodology using AES 256 is implemented in OpenEPS.
3) Session Key Material between POS and POI Device. Dynamic encryption key methodology using AES 128 is implemented in OpenEPS.

In all scenarios, there are no key custodians or intervention required by customers or resellers/integrators. Dynamic keys are generated using cryptographically secure key material, as needed.

Session key material is split between OpenEPS and SeverEPS. OpenEPS generates new session key material using dynamic material when the Cashier logs onto the POS Application. All cryptographic material will automatically be regenerated when it expires, or no greater than one year after it was created. Please refer to cryptoperiod guidelines found in NIST 800-57 Pt. 1, Rev. 4 for more information.

Encryption Key Manual Regeneration

The ServerEPS web portal allows administrators to generate new key material for a given lane (OpenEPS & POS implementation). The portal displays the date and time when the key material was generated, and provides the user with a button that re-generates the key at the lane. This feature is available with OpenEPS version 827.3 and higher.

In order to manually generate new key material:

1) Log into the ServerEPS Web Portal
2) Select Monitoring -> Store Status from the tab menu
3) Search for desired Store

4) Expand the desired Lane
5) Note the date/time stamp of the current key, and press Regenerate Key button.
6) The update process can take anywhere from 15 to 30 minutes.

In order to validate that the lane is updated correctly, confirm that the date/time stamp has been updated with a recent date/time.

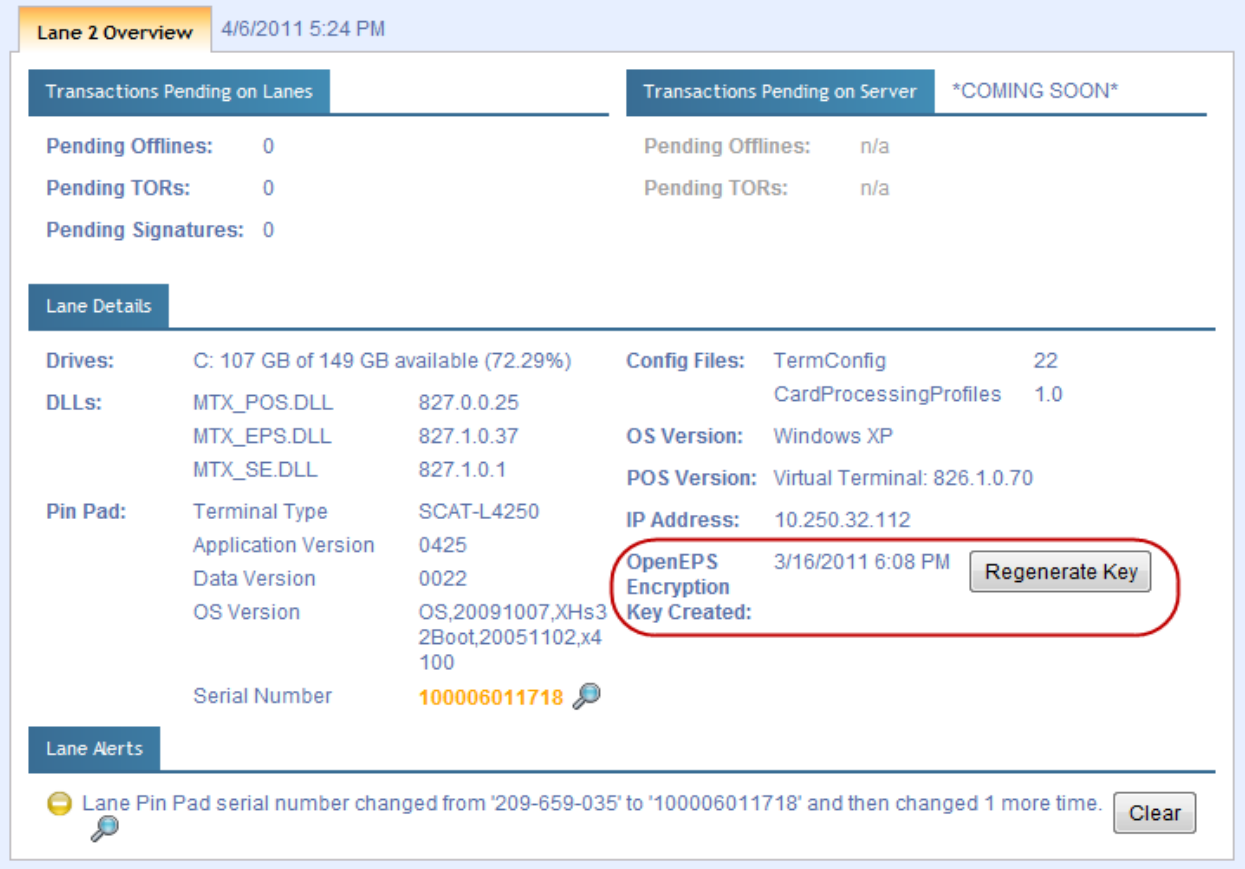


Figure 7 - ServerEPS Screen-shot for OpenEPS Key re-generation

Removal of Historical Cryptographic Material (PA-DSS 2.6)

OpenEPS uses previously validated encryption algorithms that are PCI Compliant. Therefore, there is no need to render historical cryptographic keys or cryptograms irretrievable as they are still in use by the payment application.

Set up Strong Access Controls (3.1 and 3.2)

PCI DSS "Requirement 7," requires that access to all systems in the payment processing environment be protected by various security principles, including but not limited to: need to know, role base access, and privileged user IDs. Use of generic group accounts, for the purpose of being used by more than one entity, is not permitted.

Authentication credentials are not generated or managed by OpenEPS. Instead, authentication credentials used by OpenEPS are provided by the underlying operating system which may utilize embedded or external authentication services, such as active directory, located in your environment.

To maintain PCI DSS compliance the following 11 points must be followed per PCI DSS:

1. Do not use group, shared, or generic IDs, passwords, or other authentication methods (*PCI DSS 3.2, 8.5; PA-DSS 3.2, 3.1.5*)
2. You must assign unique IDs for all user accounts. (*PCI DSS 3.2, 8.6; PA-DSS 3.2, 3.1.5*)
3. 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 discouraged 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.
    f. You must provide at least one of the following three methods to authenticate users: (*PCI DSS 8.2 / PA-DSS 3.1.4*)
        i. Something you know, such as a password or passphrase
        ii. Something you have, such as a token device or smart card
        iii. 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 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.
6. 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*)
7. 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*)
8. 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*)
9. 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:

1. You must be logged on as an administrator to perform these steps.
2. If your computer is on a domain, only your network administrator can change password policy settings.
3. 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.
4. 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.
5. In the left pane, double-click Account Policies, and then click Password Policy.
6. Double-click the item in the Policy list that you want to change, change the setting, and then click OK.
   a. Policies are as follows:
      i. Enforce Password History
      ii. Maximum Password Age
      iii. Minimum Password Age
      iv. Minimum Password length
      v. Password must meet complexity requirements
      vi. Store passwords using reversible encryption

See also: http://windows.microsoft.com/en-us/windows/change-password-policy-settings

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

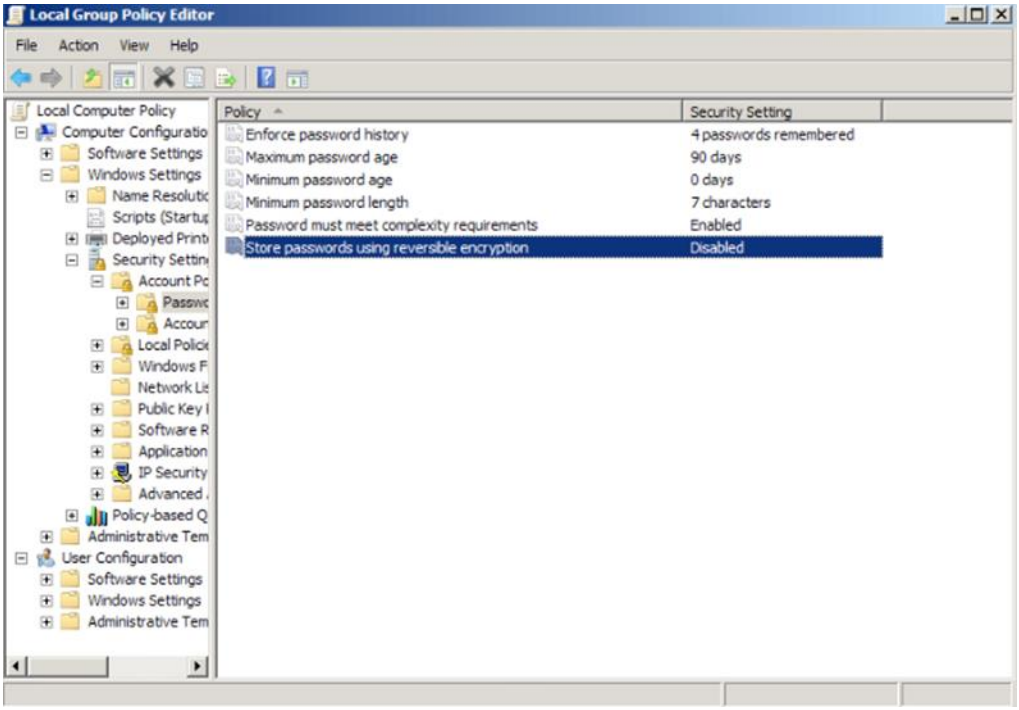


Figure 8 – Passwords, Local Group Policy Editor

Note: The recommended ***Minimum password age*** is 1 day. This will prevent the user from changing the password multiple times in one day.

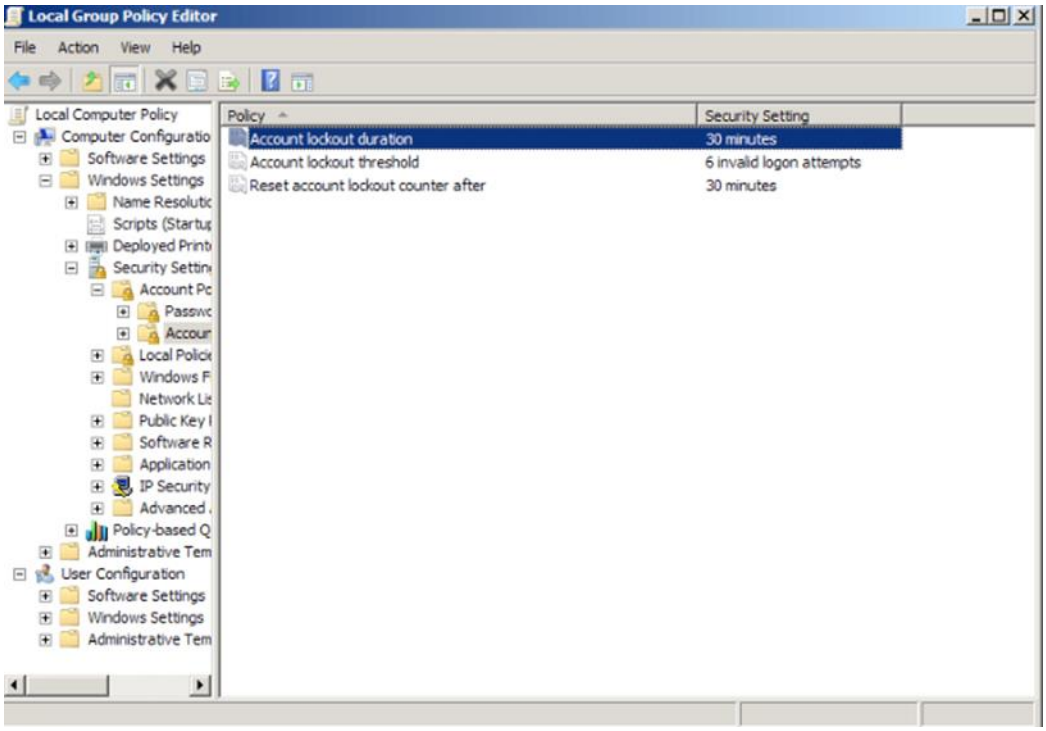


Figure 9 - Passwords, Local Group Policy Editor

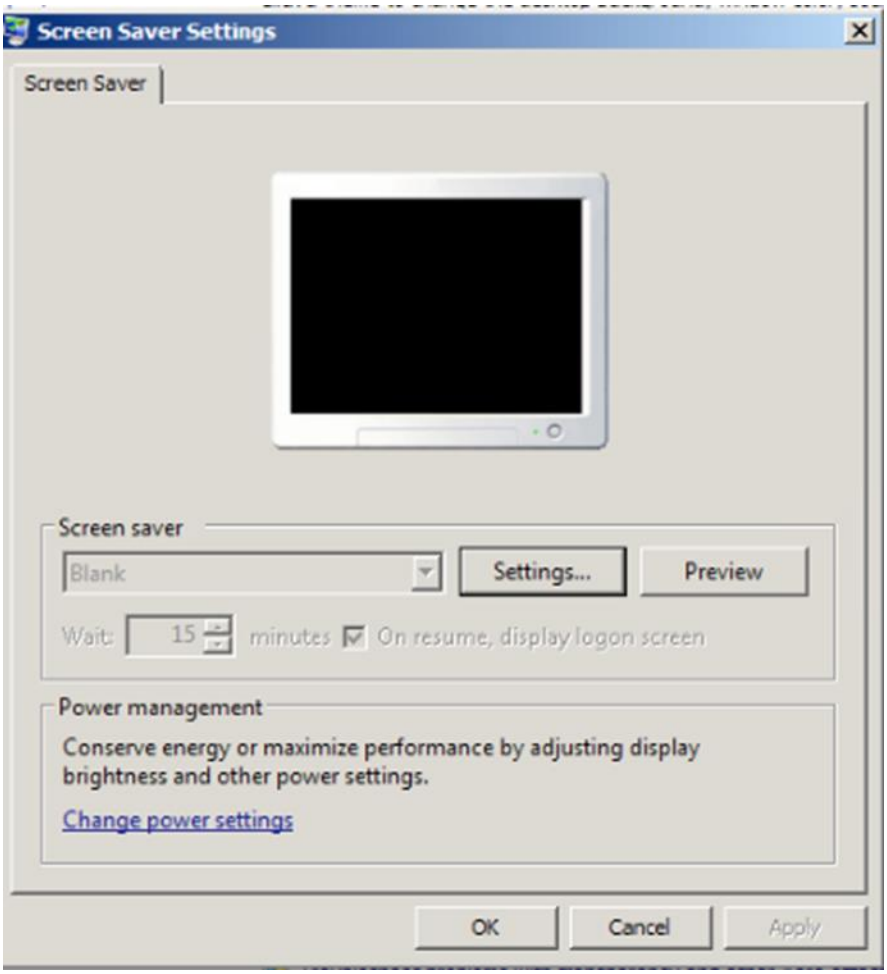


Figure 10 - Screen Saver Configuration

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 3.1: 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, etc. You can control whether each individual admin user can see credit card PAN (or only last 4).

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 CHD 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 chooses 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 - File/Folder Auditing Policy Settings 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 the following PCI DSS requirements 1.2.3, 2.1.1, and 4.1.1, as follows:

4.1.1: Industry best practices must be used to implement strong encryption for authentication and transmission of cardholder data. In example, the use of WEP as a security control is prohibited as of June 30, 2010.

1.2.3: 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. See Addendum - Firewall Configuration for more information.

2.1.1: Change wireless vendor defaults, as follows:

- 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.
- Default SNMP community strings on wireless devices must be changed. For example, on a Cisco controller:
    (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>***
    (a) The name parameter is the community name (in this case, "***public***" or "***private***").
    ii) To create a new community, enter this command: ***config snmp community create <name>***
    (1) Enter up to 16 alphanumeric characters for the name parameter.
    (2) Do not enter "***public***" or "***private***."
    iii) 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***
    iv) 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***
    v) To enable or disable this SNMP community, enter this command: ***config snmp community mode {enable | disable} name***
    vi) To save your changes, enter ***save config***.
    (1) Repeat this procedure if you still need to change the default values for a "public" or "private" community string.
- Default passwords/passphrases on access points must be changed. For CISCO devices, the following commands are used to change passwords. For more information on CISCO Aironet

Access Point configuration, see: http://www.cisco.com/c/en/us/td/docs/wireless/access_point/12-4_10b_JA/configuration/guide/scg12410b.pdf

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

- Firmware on wireless devices must be updated to support strong encryption for authentication and transmission over wireless networks.
- Other security-related wireless vendor defaults, as applicable, must be changed
- Perimeter firewalls must be installed between any wireless networks and systems that store cardholder data, and these firewalls must deny or control any traffic from the wireless environment into the cardholder data environment. See Addendum - Firewall Configuration.

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, etc., will require the following protocols and ciphers
  - HTTPS
    - Port 443
    - TLS 1.2 (see Addendum - TLS and FIPS 140-2)
  - CRL (Certificate Revocation Lists)
    - Port 80
  - OCSP (Online Certificate Status Protocol)
    - Port 80

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

OpenEPS is specifically designed to not store cardholder data in the Cardholder Data Environment, unless absolutely needed for off-line processing. Furthermore, never store cardholder data on Internet-accessible systems. Web servers and database servers must never be deployed to the same machine.

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 factors 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. While Connected Payments cloud environment utilizes two-factor authentication, it is outside the scope of the merchant's environment.

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

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

- OpenEPS is in active development. NCR 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 achieved using TLS v1.2 at NCR's website.
- 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.
- OpenEPS utilizes Code Signing and verification. If the OpenEPS binary has been tampered, the software will not execute.

As NCR is a Software 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 a relevant vulnerability has been identified, we work to develop and test a patch that helps protect OpenEPS against the 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 data flow diagrams (Hardware, ESE) for an understanding of the flow of encrypted data associated with OpenEPS.

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

OpenEPS does not allow or facilitate the sending of CHDs 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 networks 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 hosts within 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 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.
- Acquire and assistance and opinions from Subject Matter 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 2009 (with extended support)
- POSReady 7
- Windows 7
- Windows 8.1
- Windows 10

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.

For detailed information regarding installation and configuration of OpenEPS, please refer to the "*Connected Payments Installation and Configuration Guide".*

Briefly, OpenEPS installation and configuration is performed as follow:

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 does files: *.tor, Spool*, actlog*, jrnl*, Off-line data*) 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 directories specified by these local registry entries:

- “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 off-line transactions to the off-line transaction 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 – Installation Instructions

The instructions provided in this Addendum is a subset of "*Chapter 2 – Installation & Configuration"*, of the "*Connected Payments Installation and Configuration Guide, v2.31, May 2017*". That reference documents the Standard Practice for OpenEPS & Connected Payments Installation and Configuration.

Acquiring the Connected Payments Software

To install a Connected Payments payment solution, the following software packages are required:

- OpenEPS Direct required software:
  - OpenEPS to Connected Payments installation package
  - [Connected Payments Install XXX.X.Exe]
- Connectivity testing utility (Optional for testing connection between OpenEPS and the Connected Payments server)
- [ServerEPSConnectionTest.exe]
  - Connected Services Agent installation zip file

Installation User Account Information

The Installation process will require Read/Write/Modify rights to the Windows Registry and to the Program Files directory in order to create the proper registry entries and install the OpenEPS files. If the MTX_POS.DLL resides somewhere other than a Program Files sub-directory, the installer will require Read/Write/Modify rights to that folder as well.

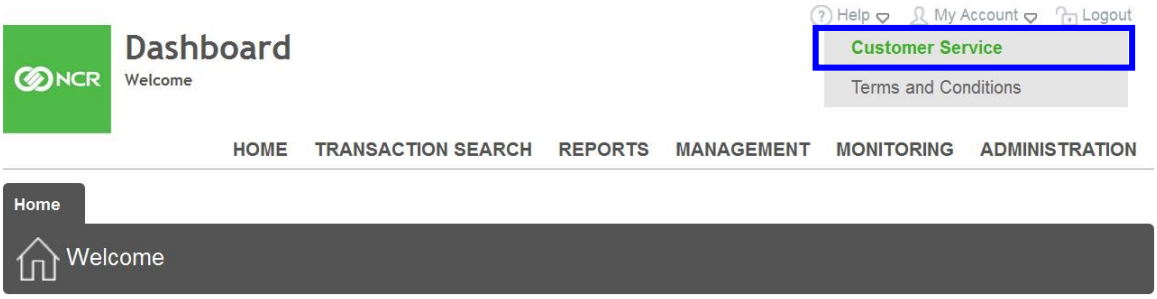
POS User Account Information

When OpenEPS runs, it runs under the same account the POS uses. See the

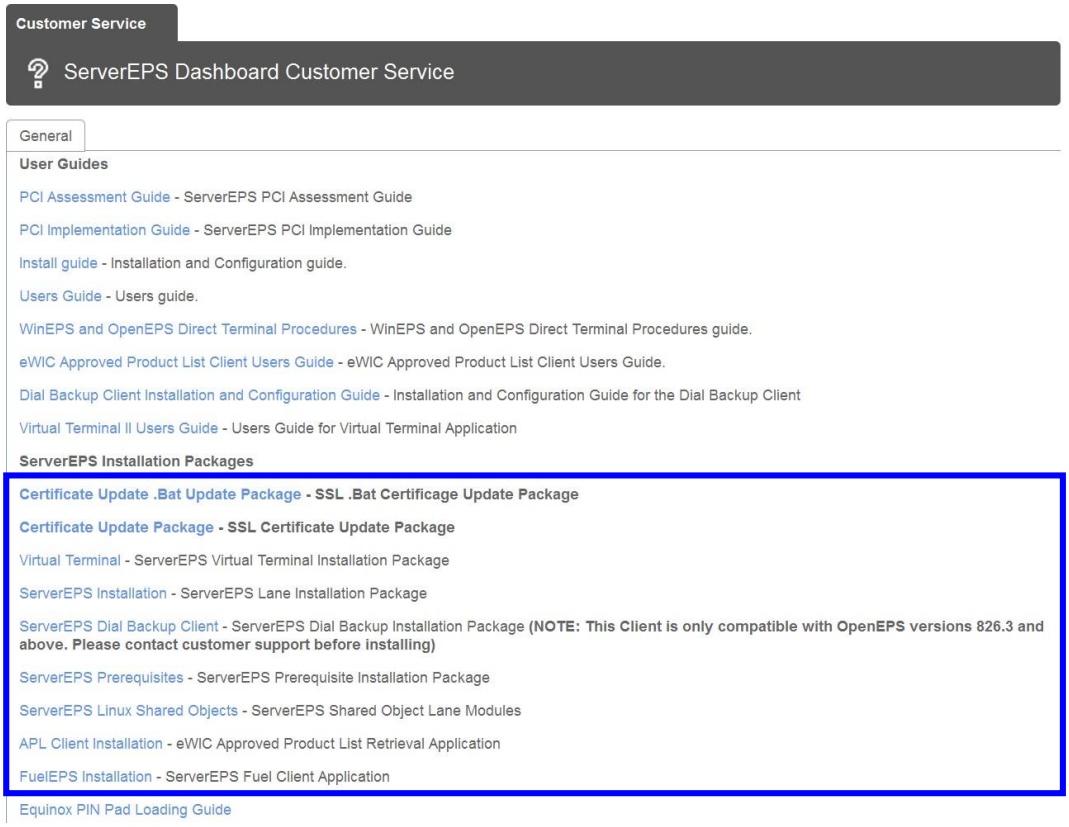
OpenEPS/POS User Account Permissions section for POS account requirements.

Connected Payments Transaction Management Portal

The above required software installation packages will either be provided, or you can download them through the online Transaction Management Portal. To reach the portal, log in through the StoreNext web page (See details in the Connected Payments User's Guide). Once you reach the Transaction Management Portal, select the Customer Service link.



This link will bring you to a page of convenient downloadable material, including Release Notes, Implementation Guides, and the installation packages. Download the installation packages that you need by clicking on their link and selecting Save.



OpenEPS Installation and Configuration

This section covers installing and configuring OpenEPS.

It is assumed that the Point of Sale software that will be used in the payments environment has already been installed by this point. If it has not, install the POS system before proceeding.

Installation Process

The following steps are required for both brand new store installations that are installing Connected Payments for the first time, and store locations with pre-existing WinEPS installations that are upgrading to Connected Payments.

1. Acquire Company and Store Number
2. Install Virtual Terminal (Optional)
3. Install OpenEPS to Connected Payments Package
4. Test Connected Payments Connectivity

Acquire Company and Store Number

The Company Number and Store Number are used by Connected Payments to identify transactions and to route payments to the correct merchant.

If you have not already acquired the Company Number and Store Number that will be used in the store you are installing, contact StoreNext. StoreNext will assign you a company number as part of the sign-up process for the Connected Payments service. You will assign a unique store number of up to 8 digits in length for each of your store locations, and submit those numbers to StoreNext.

The Company Number and Store Number are used during the installation and configuration of the system, and should be on hand before the installation process is started.

Install Virtual Terminal (Optional)

Virtual Terminal 2 (VT2) is a lightweight Windows software application that can be used with the

OpenEPS Direct payments solution to process transactions, similar to a POS system. The Virtual Terminal installer package can be acquired from StoreNext and can be used free-of-charge as part of OpenEPS Direct.

Since Virtual Terminal is a POS program it may not be installed on the same PC that other POS software is running on. It may be installed on the same machine on which the Dial Backup Client is installed (providing the Dial Backup Client is not installed on a POS lane).

To install Virtual Terminal, follow the instructions in the Virtual Terminal User's Guide. It is important that Virtual Terminal be installed prior to performing the Install OpenEPS to Connected Payments Package step; if Virtual terminal is installed second, the installation will overwrite required OpenEPS Direct files and settings required to connect successfully.

If you find that you have already installed the Install OpenEPS to Connected Payments Package on a machine and wish to add Virtual Terminal, simply install Virtual Terminal and then perform the Install OpenEPS to Connected Payments Package installation again.

Install OpenEPS to Connected Payments Package [Windows]

(This section covers the installation of OpenEPS on a Windows PC; for Linux skip to the Linux section.)

The OpenEPS Direct installation consists of an updated OpenEPS that sends transaction information directly to the Connected Payments data centers.

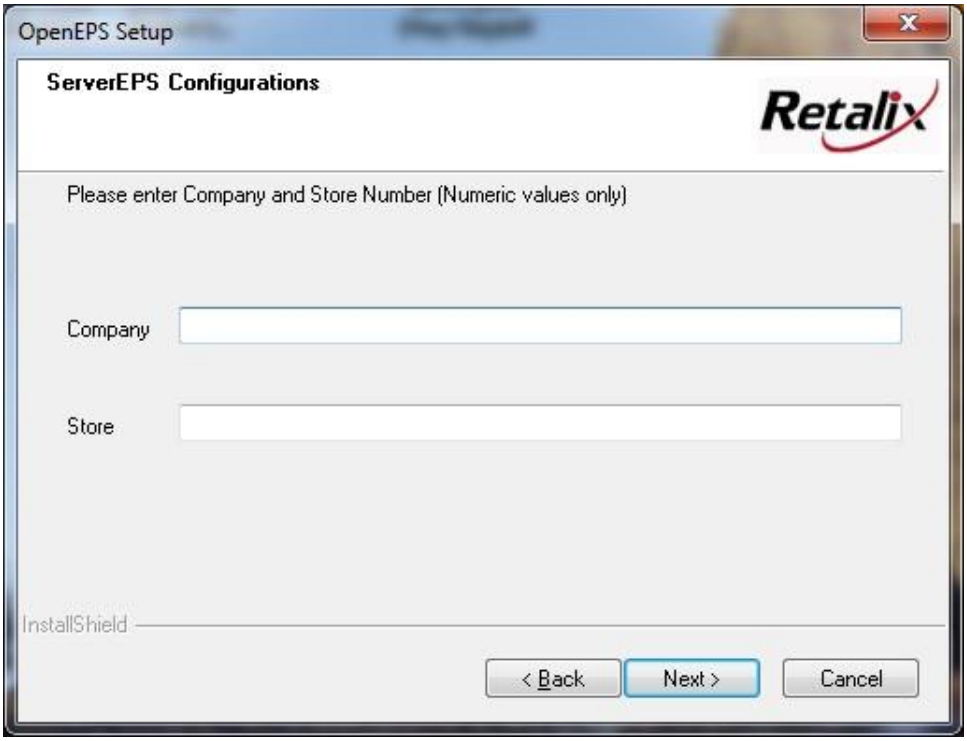
Information on installing OpenEPS from the command line is located in the (Command Line) Silent Install Option for OpenEPS to Connected Payments Package section.

Follow the steps below to apply the new OpenEPS Direct to Connected Payments installation:

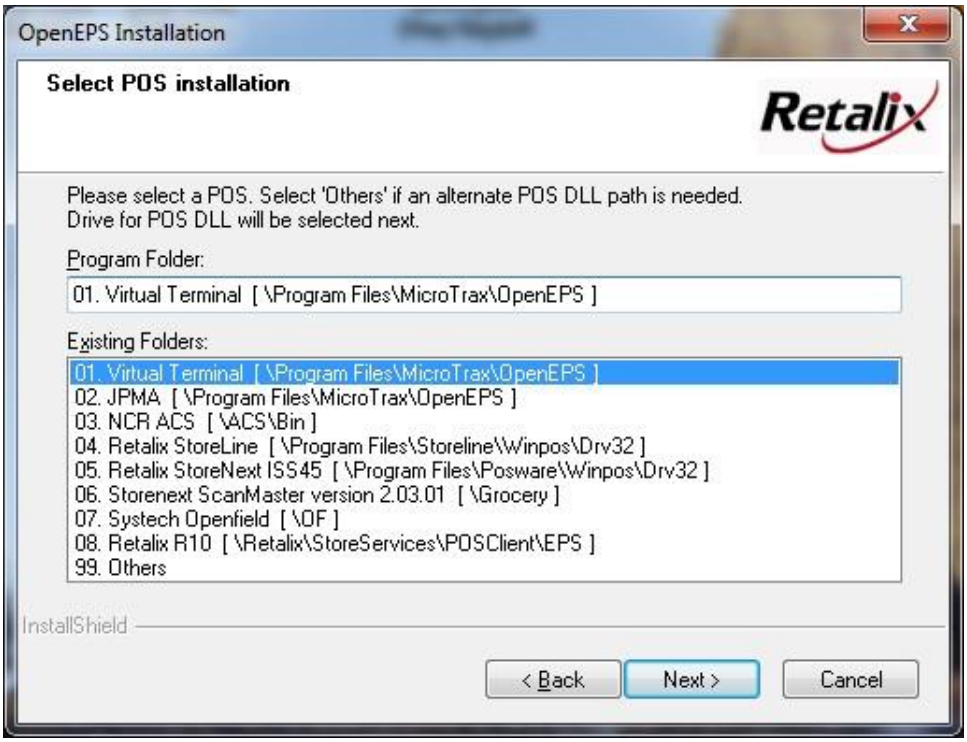
1. Copy the "Connected Payments Install XXX.X.Exe" file (where XXX.X is the current version number) onto the POS lane that you are installing.
2. If using the **CPInstall.ini** file, place the CPInstall.ini file into the root of C: on the computer that you are installing OpenEPS on. This file will provide some of the data that the installation requests, and thus you may not see all of the screen prompts depicted in this section.
3. Double click the executable to run the installation.
4. Read the licensing agreement and then click Yes to accept.

5. Enter the Connected Payments Company Number and Store Number. Click Next to continue.

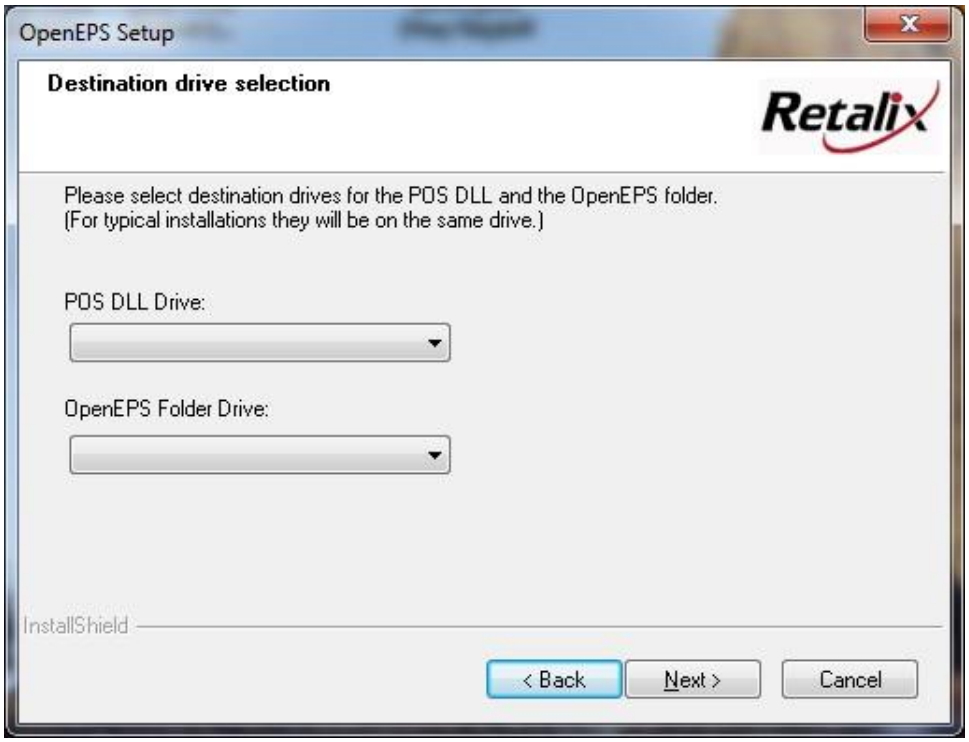
the company number to enter on the [illegible] have an expected length of [illegible] six digits. OpenEPS Setup [illegible] screen is your StoreNext company number plus 100000.



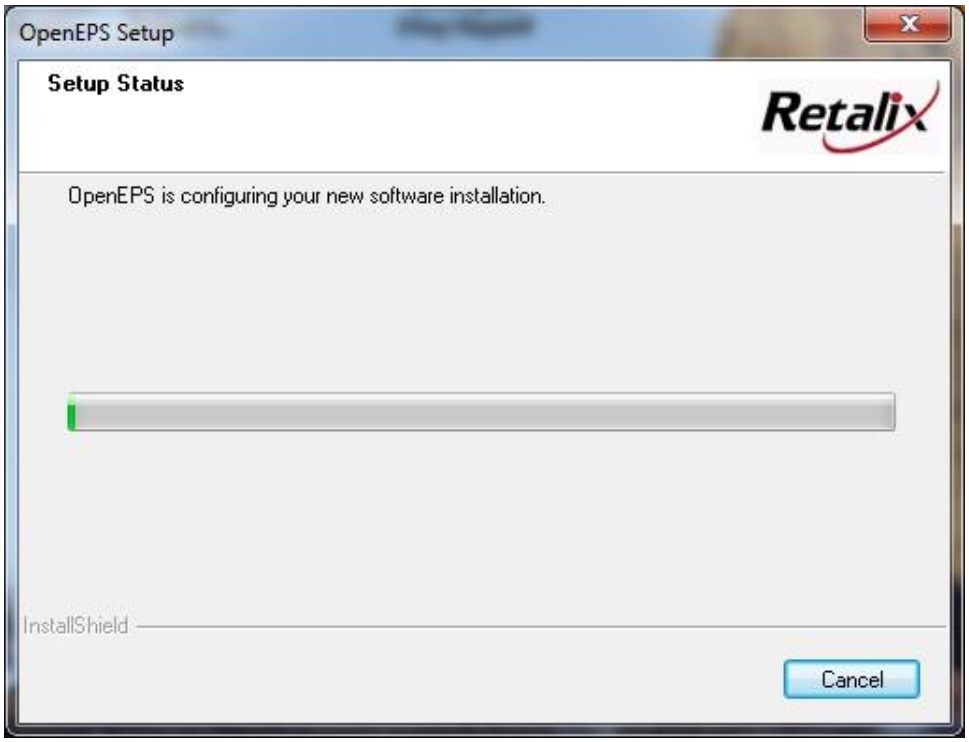
6. Select your POS system from the list, and hit Next.



7. Use the dropdown boxes to select the drive locations where the POS DLL, and the OpenEPS folder will be installed. Select the drive on which the POS is installed. Only drives that exist will be displayed. Click Next to continue.



OpenEPS will proceed to install all the required components for the OpenEPS Direct solution.



8. Repeat this installation procedure for each POS lane.

Verify POS System Clock Time & Date

As part of the installation process it is important to verify that the POS system time & date are accurate. The certificates used to support SSL connectivity use the system time as part of the validation procedure; if the POS system clock is significantly off or the date is inaccurate, transactions may not process.

(Command Line) Silent Install Option for OpenEPS to Connected Payments Package

A silent installation is an install where are the parameters that the standard install would normally request of the user are instead supplied by command line. This option can be used instead of the standard install, and is useful for aiding in automating the install process.

The command line installation can be performed instead of the standard install. The command line parameters make it easy to install OpenEPS using a batch file.

Note: The **CPInstall.ini** file is not loaded during a silent install; all appropriate information should be added view the command line switches instead.

Using command line parameters, it is possible to install OpenEPS without needing to answer prompts during the installation. The following is a list of current switches and their description:

Command-Line Switch	Description
/a/s	Silently installs OpenEPS
-c OR /c	Company Number This parameter must be entirely numeric. ▪ Connected Services company numbers have an expected length of six digits; the company number to enter on the OpenEPS Setup screen is your StoreNext company number plus 100000.

<table>
<tr><th>Command-Line Switch</th><th>Description</th></tr>
<tr><td>-i OR /i</td><td>Path to Install the MTX_POS.DLL file.
This switch allows you to specify a path for where the MTX_POS.DLL file will be placed. Use either this switch or the /t switch, but not both as they serve the same purpose.

When specifying the directory path, be sure to use quotation marks if spaces are included in any of the directory names. Example: /i”C:\Program Files\MicroTrax\OpenEPS”</td></tr>
<tr><td>-h OR /h</td><td>Host IP address
This is the IP address of the machine the Dial Backup Client is running on. This parameter only need to be included if you are using the Dial Backup Client for dial backup.</td></tr>
<tr><td>-o OR /o</td><td>Store Number
This parameter must be entirely numeric.</td></tr>
<tr><td>-p OR /p</td><td>Host Port
This is the port the Dial Backup Client will be opening for connections. This parameter is only required if the port has been changed from its default value of 6201.</td></tr>
<tr><td>-t OR /t</td><td>POS Type
This switch is used to determine where the MTX_POS.DLL file will be placed. Use either this switch or the /i switch, but not both as they serve the same purpose.

The list of valid POS types are:
<table>
<tr><th>Switch #</th><th>POS Type</th><th>Install Folder</th></tr>
<tr><td>▪ 01</td><td>Virtual Terminal</td><td>\Program Files\Microtrax\OpenEPS</td></tr>
<tr><td>▪ 02</td><td>JPMA</td><td>\Program Files\Microtrax\OpenEPS</td></tr>
<tr><td>▪ 03</td><td>NCR ACS</td><td>\ACS\Bin</td></tr>
<tr><td>▪ 04</td><td>Retalix Storeline</td><td>\Program Files\Storeline\Winpos\Drv32</td></tr>
<tr><td>▪ 05</td><td>Retalix StoreNext ISS45</td><td>\Program Files\Posware\Winpos\Drv32</td></tr>
</table></td></tr>
</table>

Command-Line Switch	Description

▪ 06	Storenext Scanmaster V2	\Grocery
▪ 07	Systech Openfield	\OF
▪ 08	Retalix R10	\Retalix\StoreServices\POSClient\EPS
▪ 99	Others	[User Defined]

If a parameter is entered incorrectly the installation process may fail or abort.

Example command line:

```
"Connected Payments Install 824.0.exe" -a -s –h192.168.0.1 -c12345 o54321 -
i"C:\Program Files\Microtrax\OpenEPS”
```

The command line may be used part of a batch file.

Test Connected Payments Connectivity (Optional)

After the OpenEPS installation has been completed, the connectivity testing utility can be used to determine if the POS lane can reach the Connected Payments server for secure transaction processing and that the company and store are properly setup on those servers.

Follow the steps below to install the testing software and verify connectivity.

1. Copy ServerEPSConnectionTest.exe application onto the local POS lane in a convenient location (a specific folder location is not required).
2. Run ServerEPSConnectionTest.exe by double clicking it.

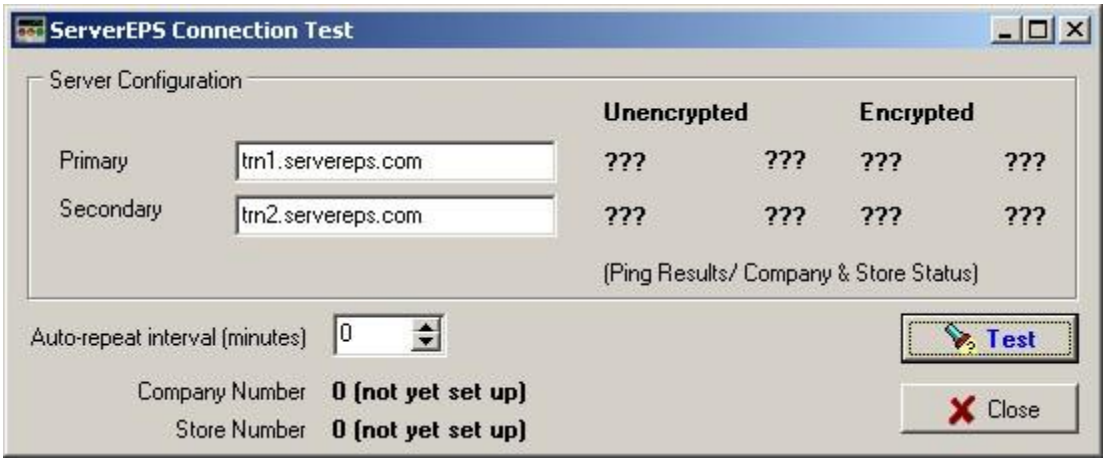
3. Verify the Company Number and Store number in the lower left are correct.
4. Click the Test button at the bottom right to check connectivity.

Successful Test

- An indication of SUCCESS under Encrypted is confirming that the POS has the necessary access to the secure Connected Payments Servers.
- An indication of OK confirms that the company and store number shown on the application are setup on the secure Connected Payments Servers.

Failed Test

- An indication of FAILED under encrypted indicates that the POS lane does not have connectivity to the Connected Payments servers
- Contact your network support and tell them your lane needs access to your Broadband Internet connection.



- An Indication of “0 (not yet set up)” under company and Store Number indicates that the required registry settings are not setup properly or at all on this local PC.
- Verify user currently logged into PC has rights to write to the Windows registry
- Please run the Connected Payments OpenEPS Installation on the POS Lane

- In the event of continued failures please contact support and indicate that the proper registry settings are not being written or read from the Windows registry.

Addendum – Addressing Inadvertent Capture of PAN

Addressing Inadvertent Capture of PAN on Windows 7

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:

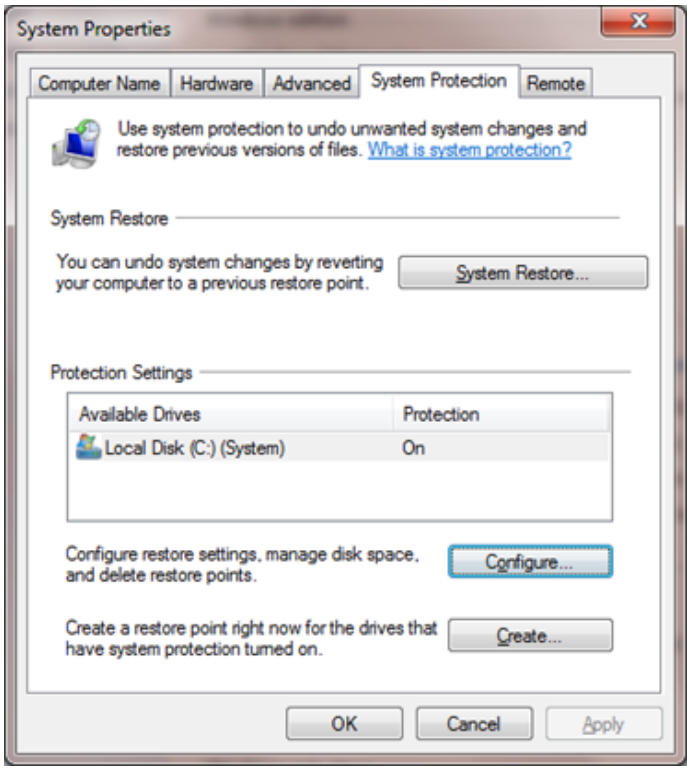


Figure 11 – Inadvertent capture of PAN: Windows 7

- Select Configure, the following screen will appear:

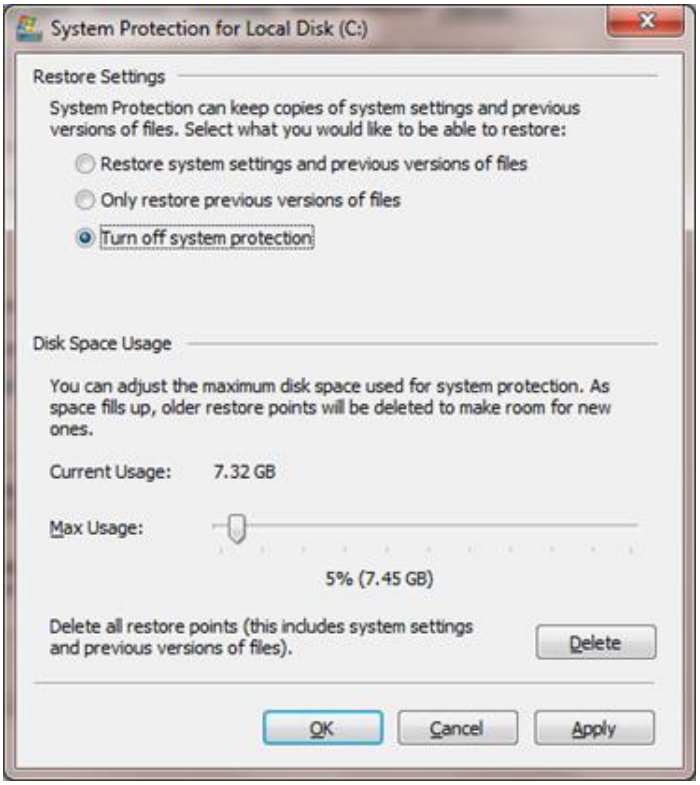


Figure 12 – Inadvertent capture of PAN: Windows 7

- 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

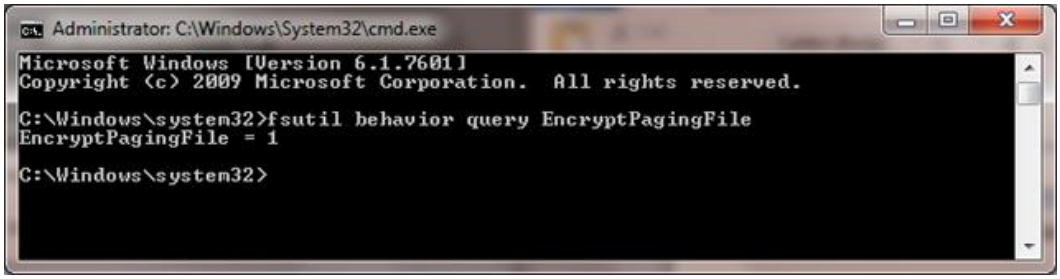


Figure 13 – Inadvertent capture of PAN: Windows 7

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

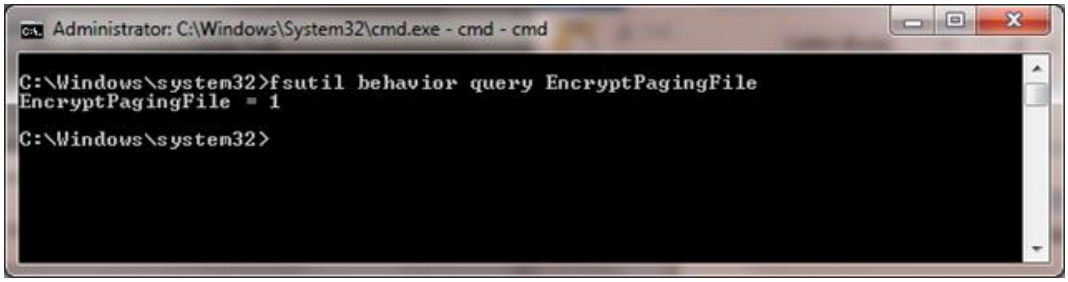


Figure 14 – Inadvertent capture of PAN: Windows 7

- 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

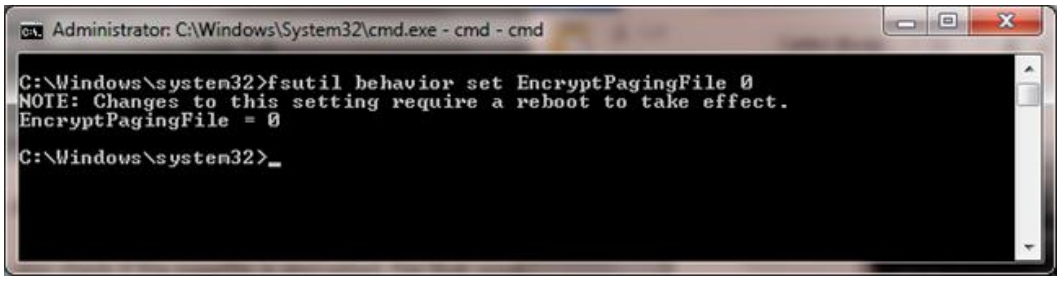


Figure 15 – Inadvertent capture of PAN: Windows 7

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

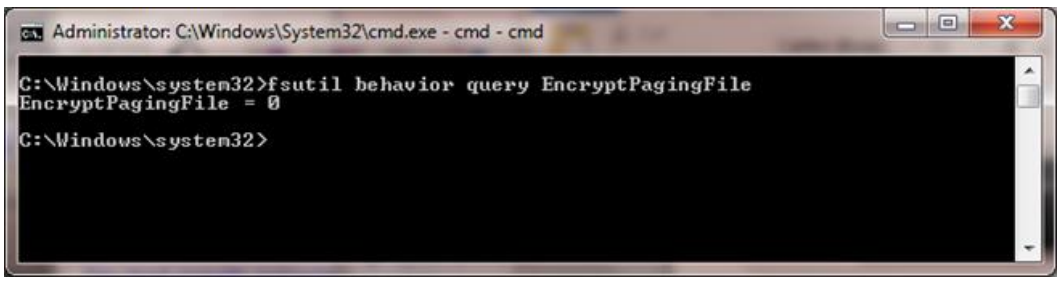


Figure 16 – Inadvertent capture of PAN: Windows 7

- 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 (CHD/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

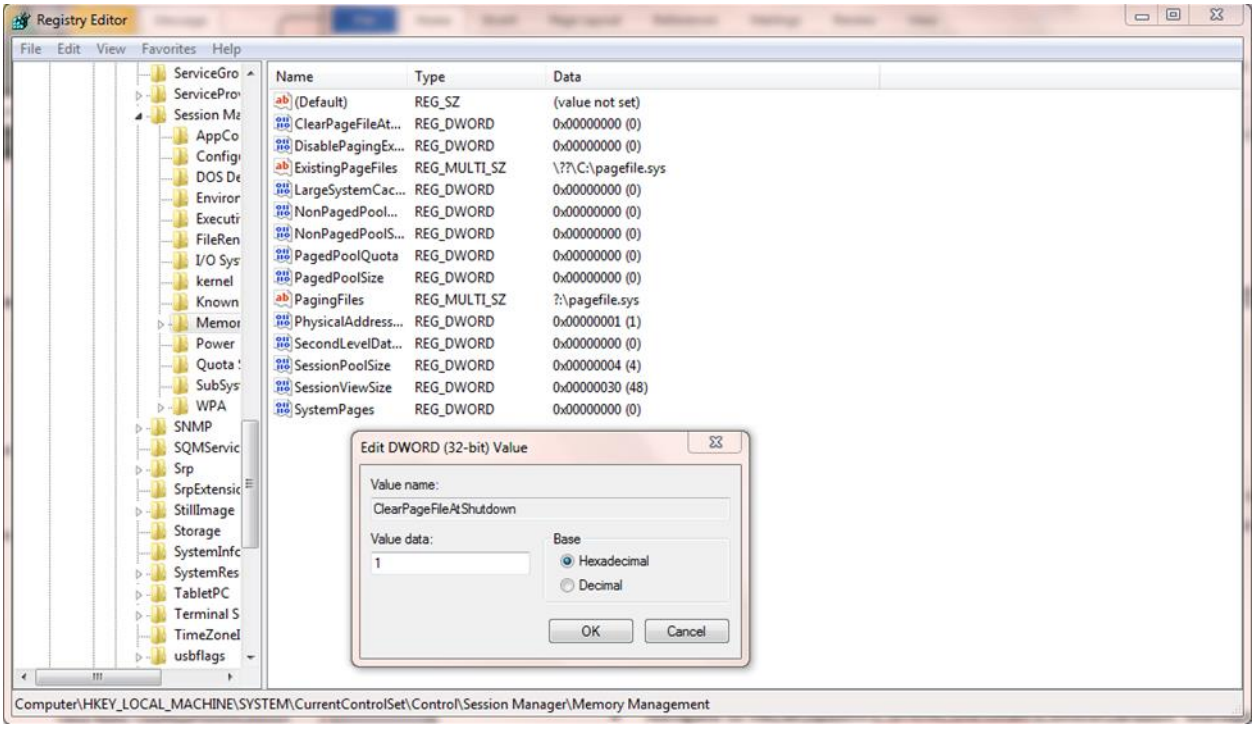


Figure 17 – Inadvertent capture of PAN: Windows 7

- 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:

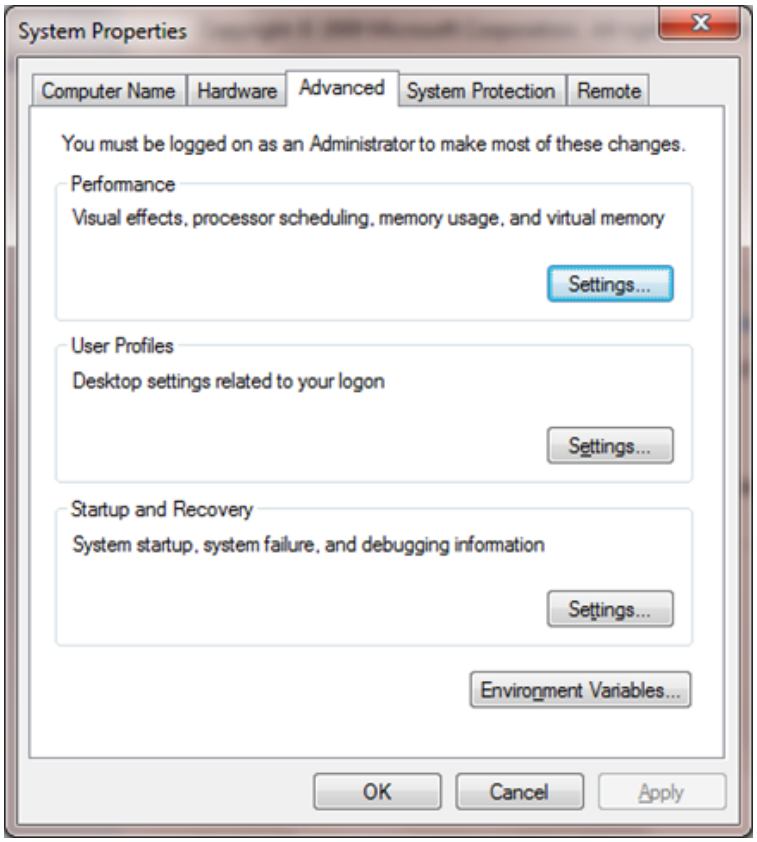


Figure 18 – Inadvertent capture of PAN: Windows 7

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

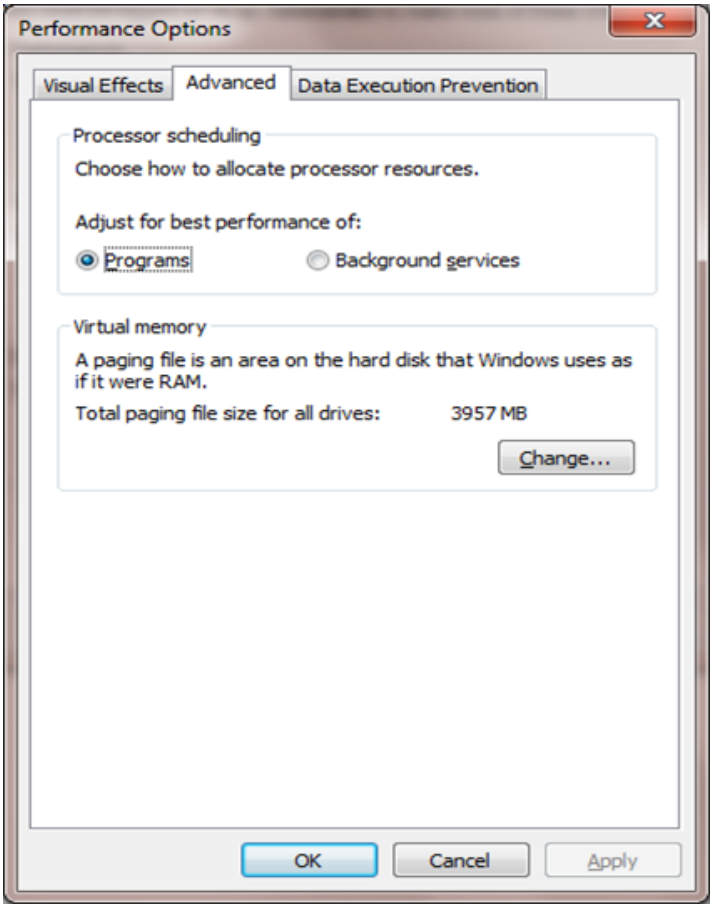


Figure 19 – Inadvertent capture of PAN: Windows 7

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

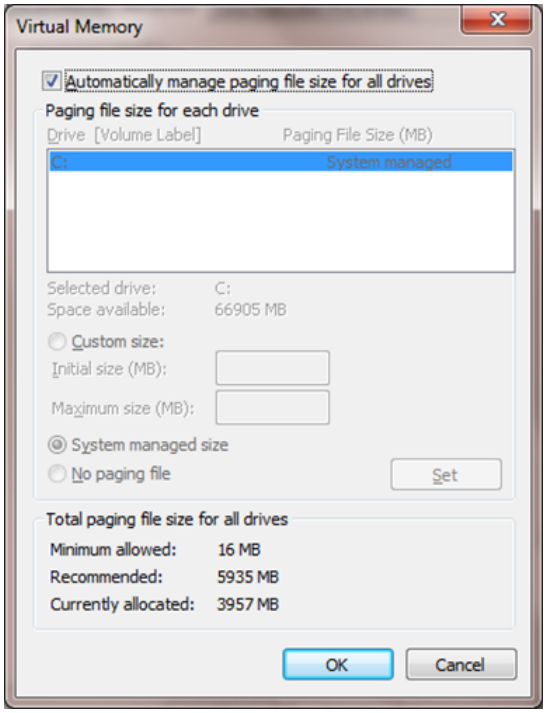


Figure 20 – Inadvertent capture of PAN: Windows 7

- 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”

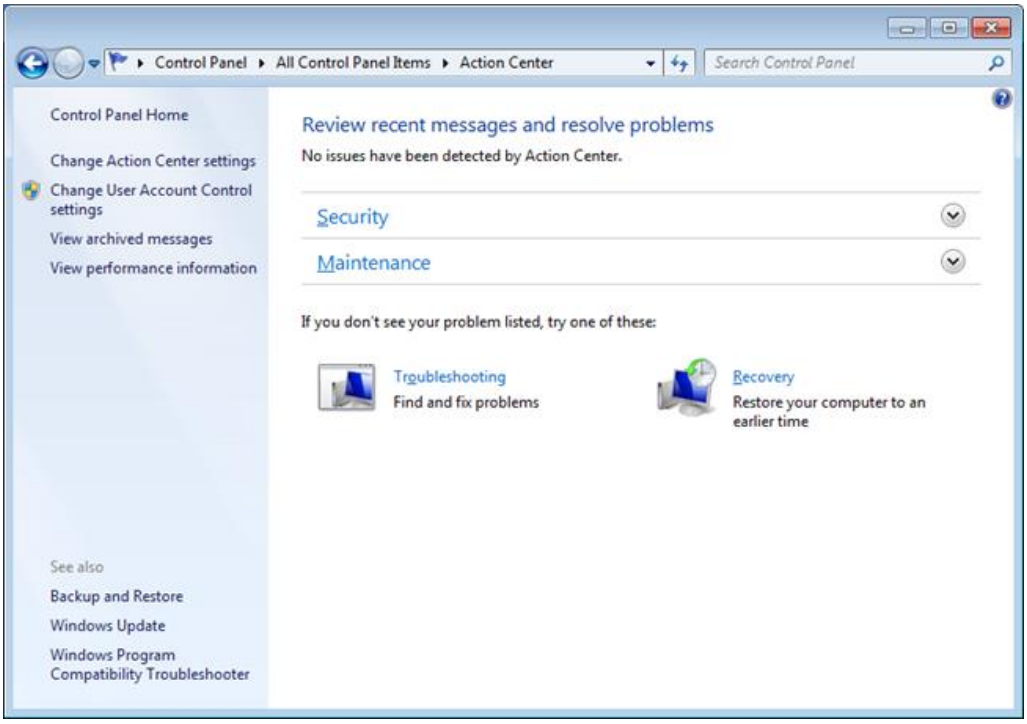


Figure 21 – Inadvertent capture of PAN: Windows 7

- Select “Problem Reporting Settings”

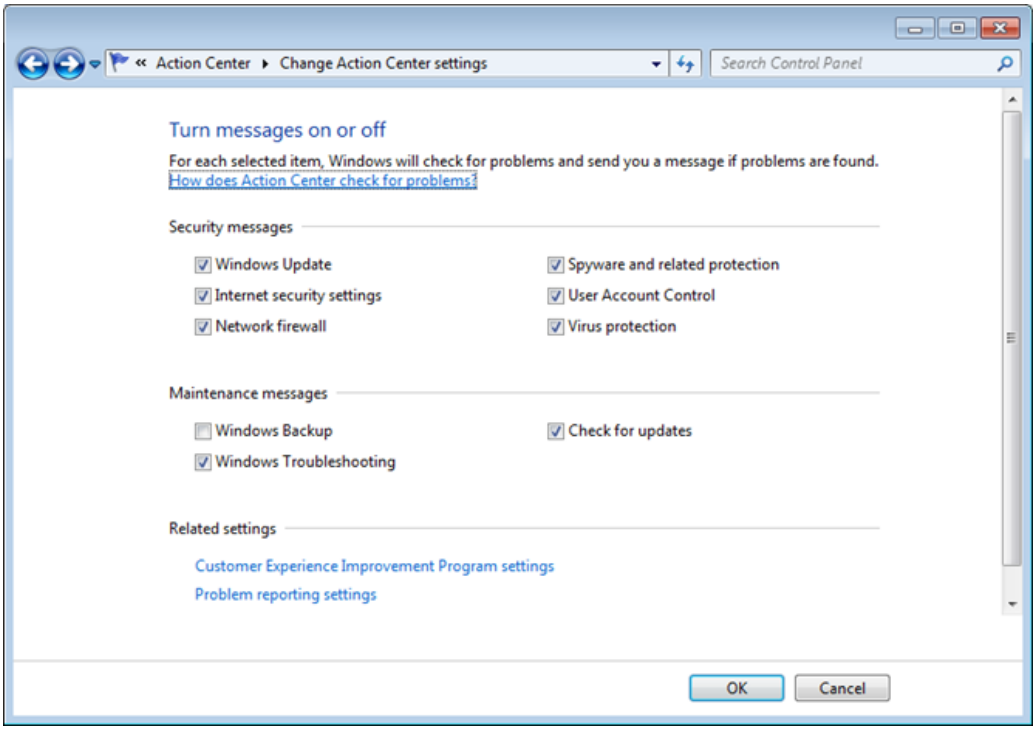


Figure 22 – Inadvertent capture of PAN: Windows 7

- Select “Never Check for Solutions”

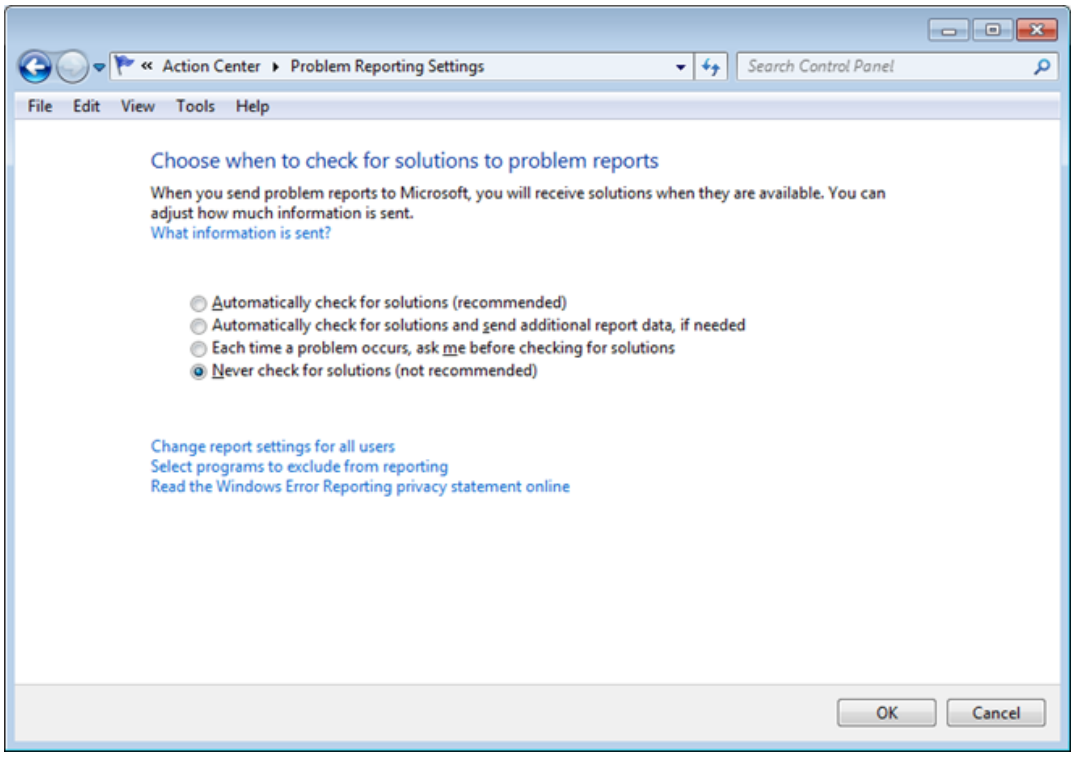


Figure 23 – Inadvertent capture of PAN: Windows 7

Addressing Inadvertent Capture of CHD on Windows 8.1

Disabling System Restore – Windows 8.1

- Right Click on Computer > Select “Properties”:

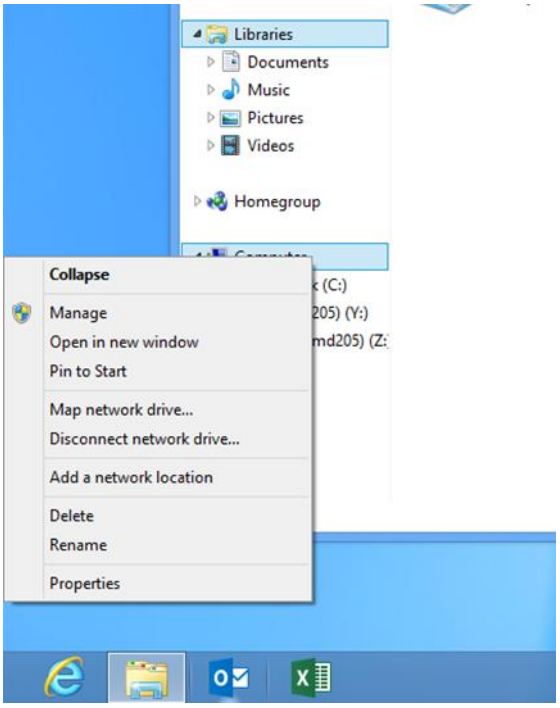


Figure 24 – Inadvertent capture of PAN: Windows 8

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

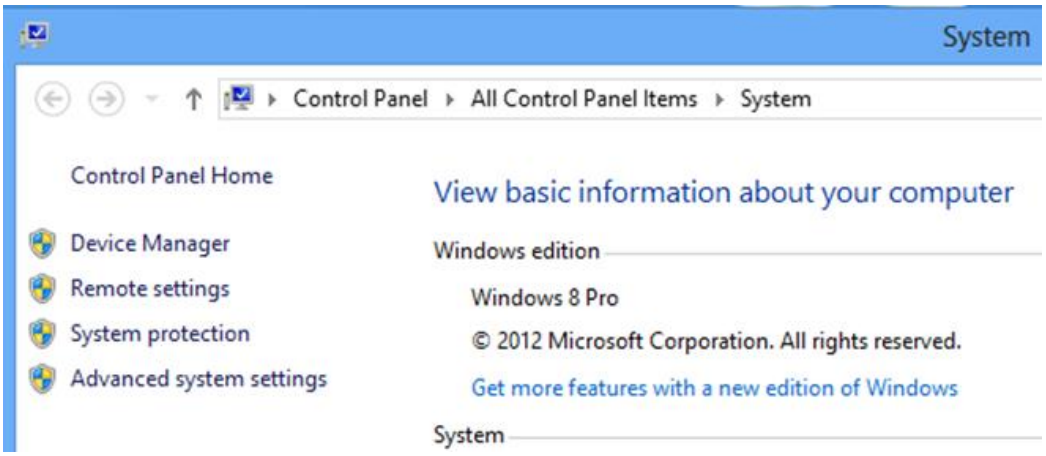


Figure 25 – Inadvertent capture of PAN: Windows 8

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

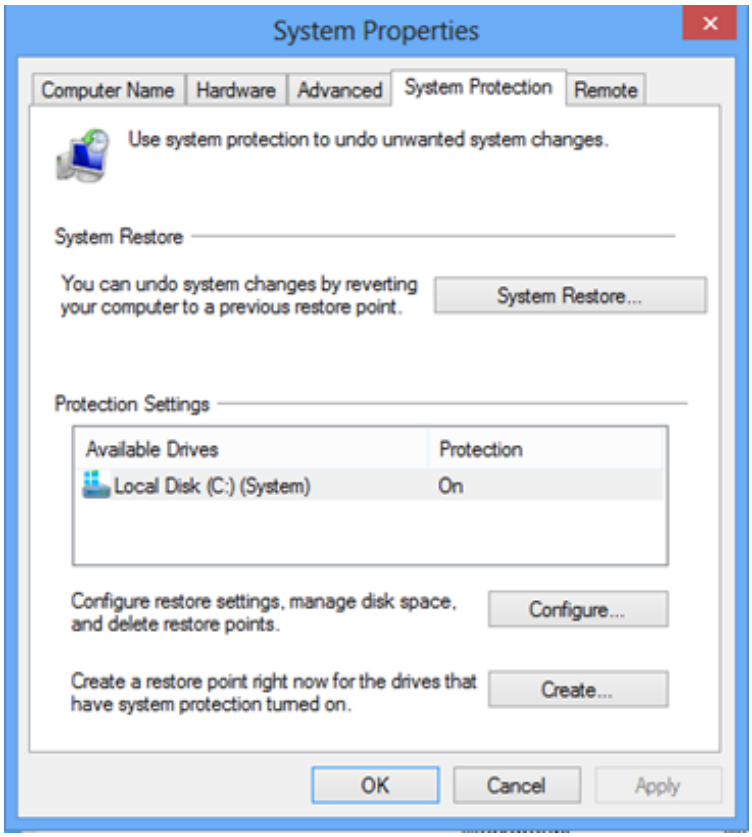


Figure 26 – Inadvertent capture of PAN: Windows 8

- Select Configure, the following screen will appear:

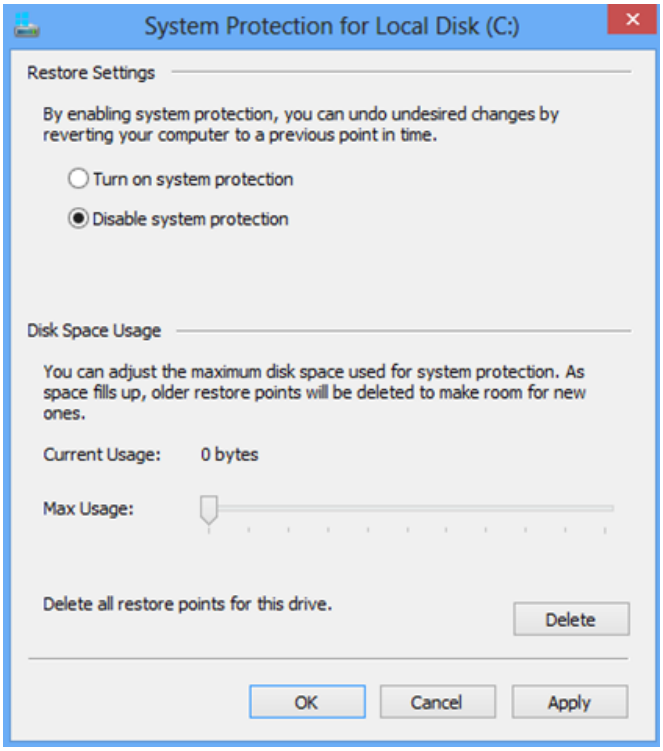


Figure 27 – Inadvertent capture of PAN: Windows 8

- 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 8.1

* 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”

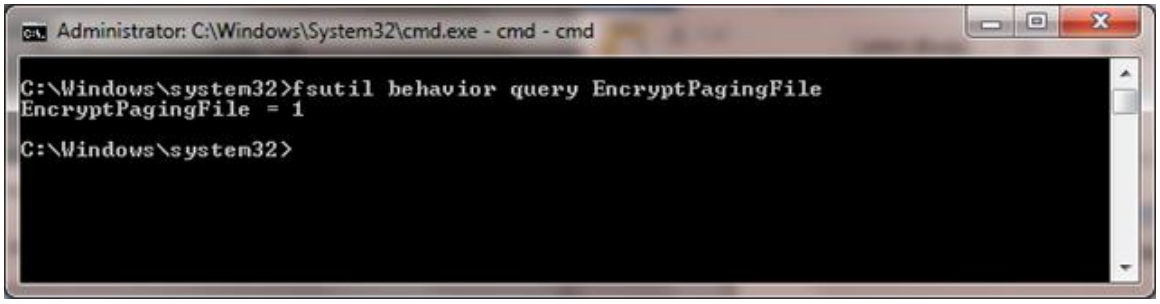


Figure 28 – Inadvertent capture of PAN: Windows 8

- 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

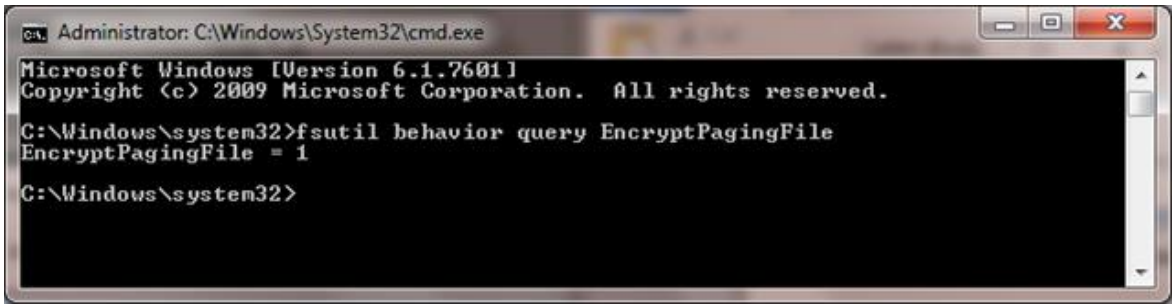


Figure 29 – Inadvertent capture of PAN: Windows 8

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

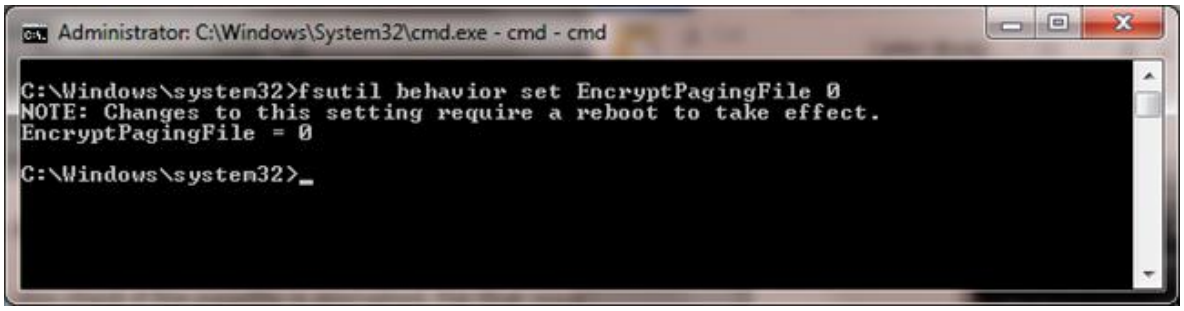


Figure 30 – Inadvertent capture of PAN: Windows 8

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 (CHD/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

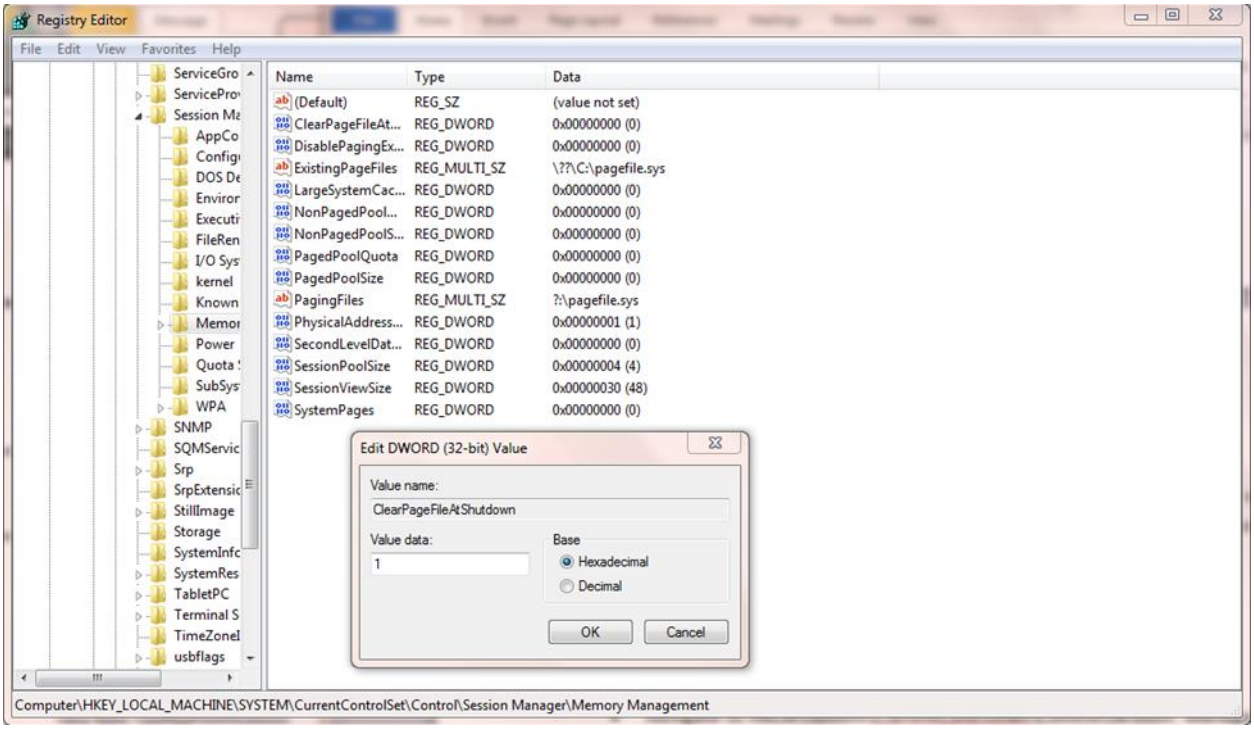


Figure 31 – Inadvertent capture of PAN: Windows 8

- 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 8.1

- Right Click on Computer > Select “Properties”:

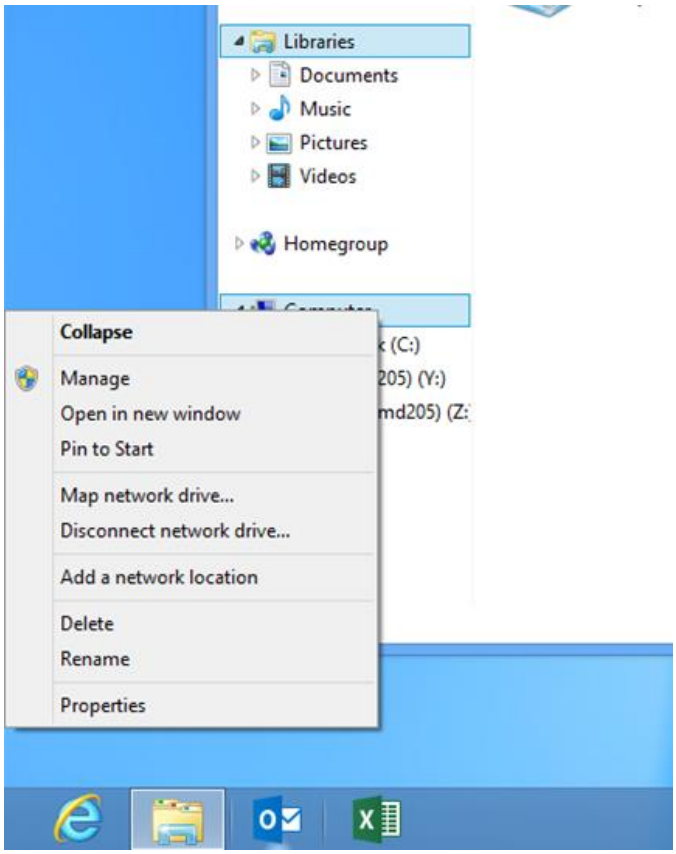


Figure 32 – Inadvertent capture of PAN: Windows 8

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

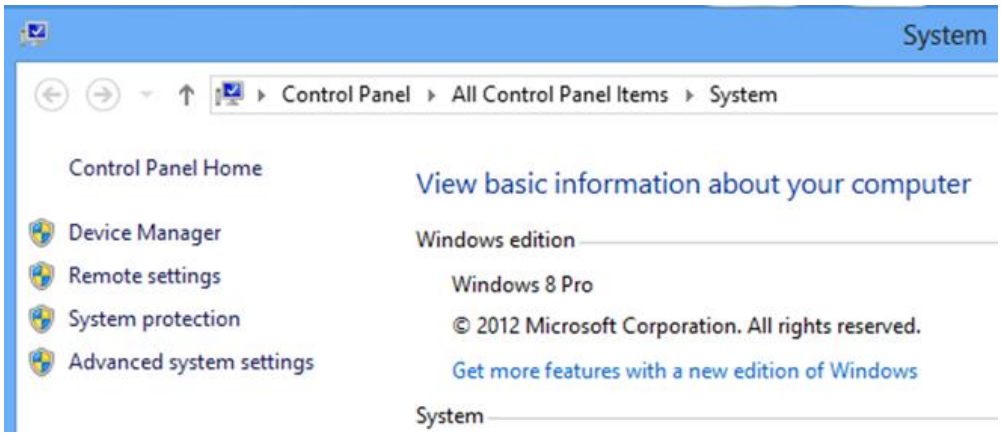


Figure 33 – Inadvertent capture of PAN: Windows 8

Select the “Advanced” tab:

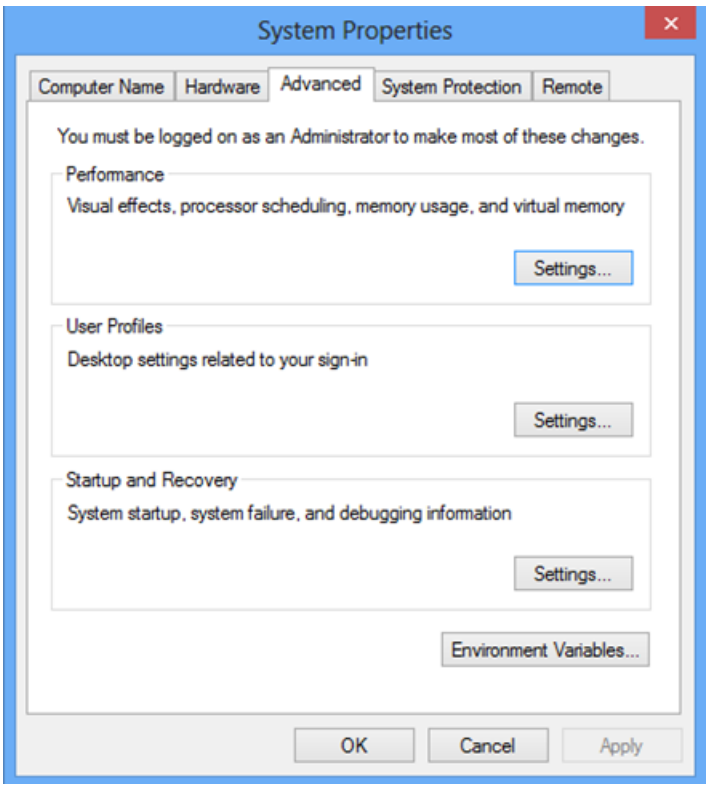


Figure 34 – Inadvertent capture of PAN: Windows 8

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

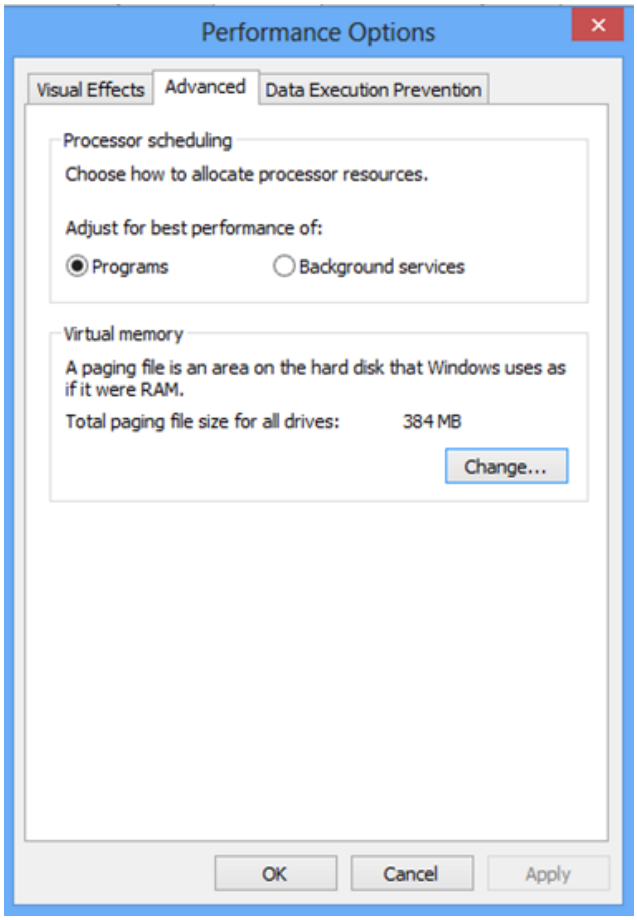


Figure 35 – Inadvertent capture of PAN: Windows 8

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

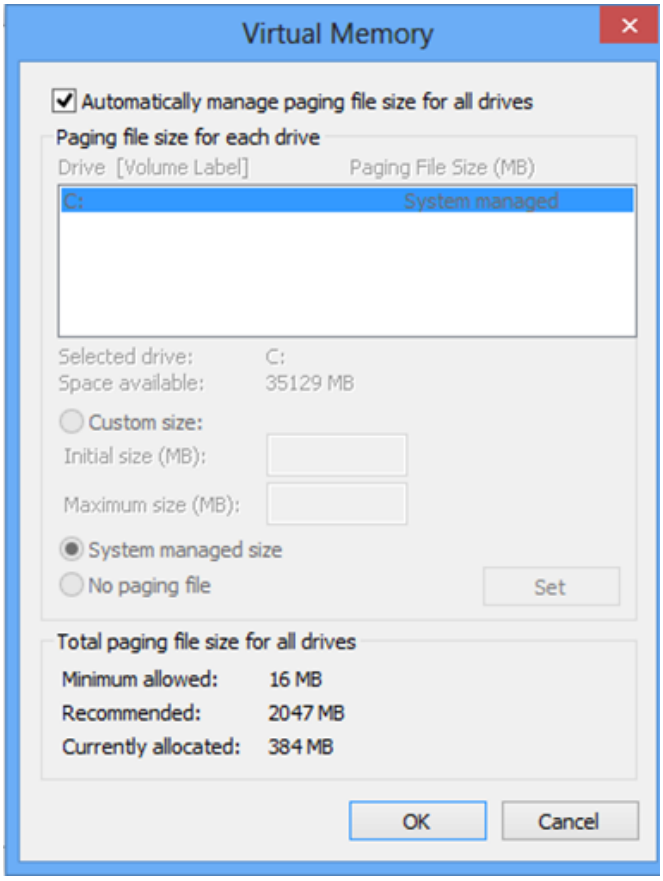


Figure 36 – Inadvertent capture of PAN: Windows 8

- 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”, 3 times
- You will be prompted to reboot your computer.

Disabling Windows Error Reporting – Windows 8.1

- 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”:

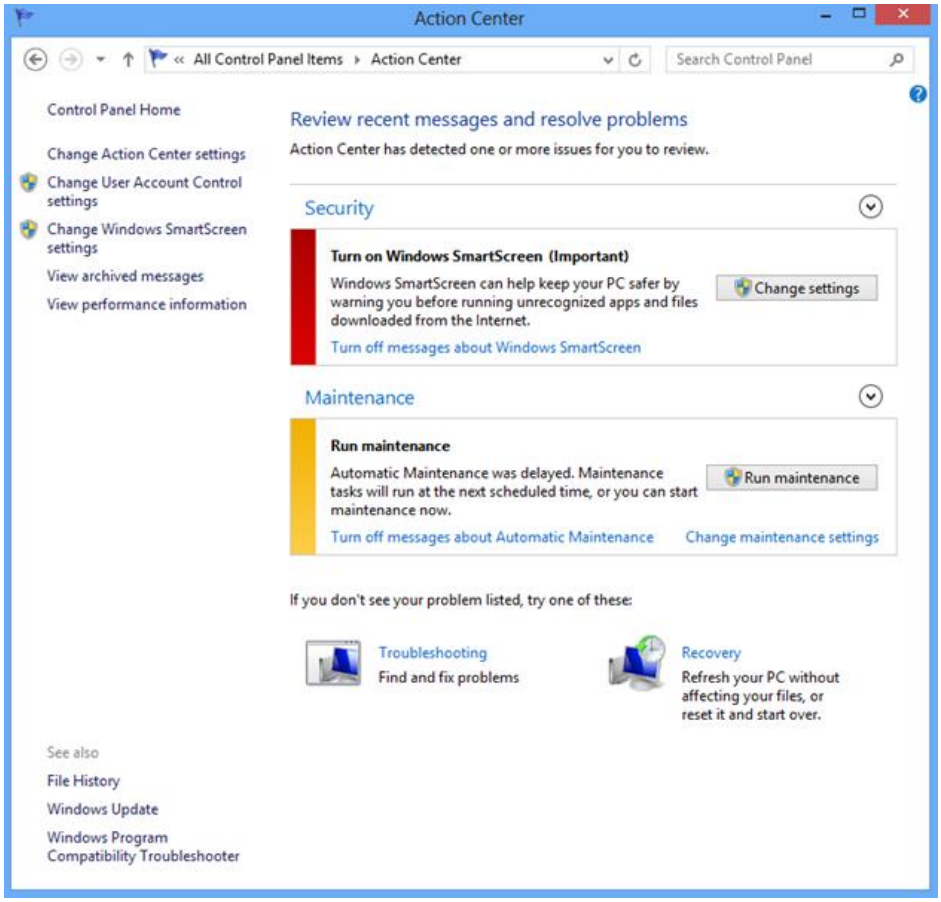


Figure 37 – Inadvertent capture of PAN: Windows 8

- Select “Problem Reporting Settings”:

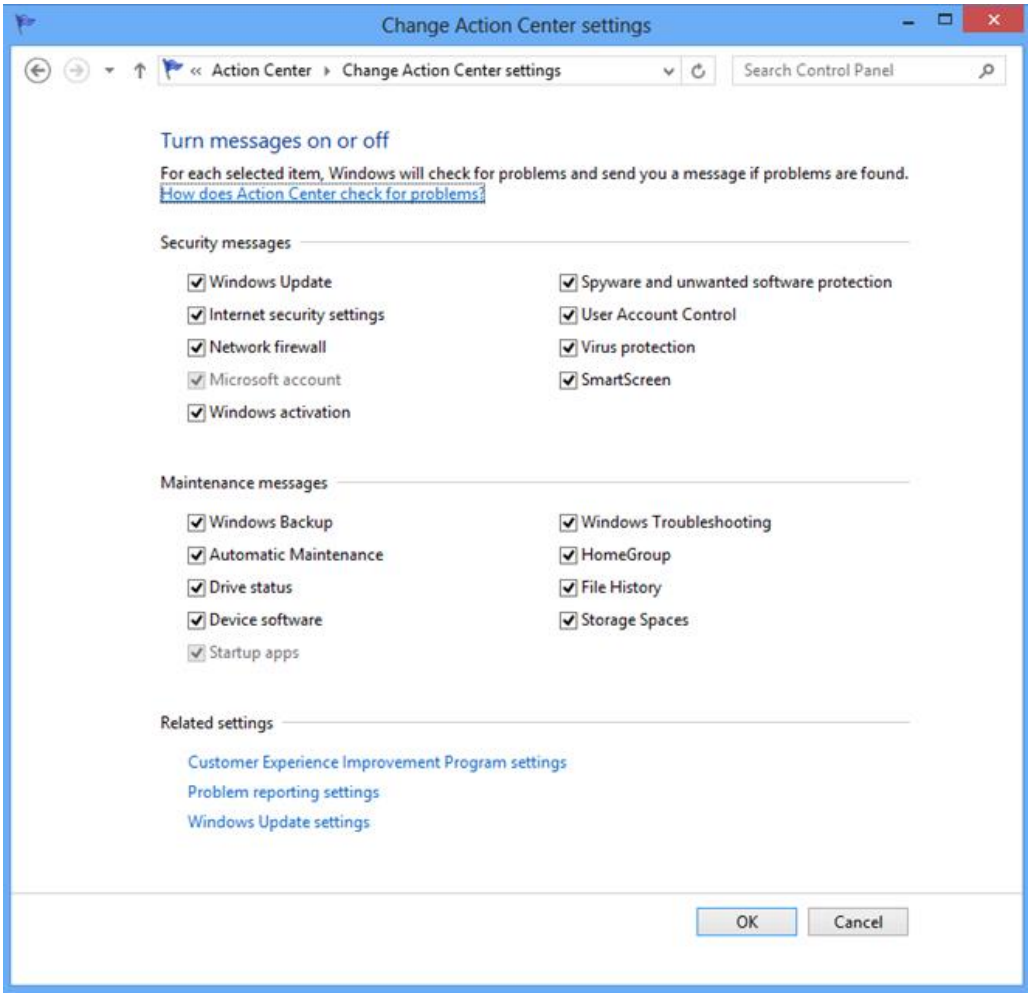


Figure 38 – Inadvertent capture of PAN: Windows 8

- Select “Never Check for Solutions”:

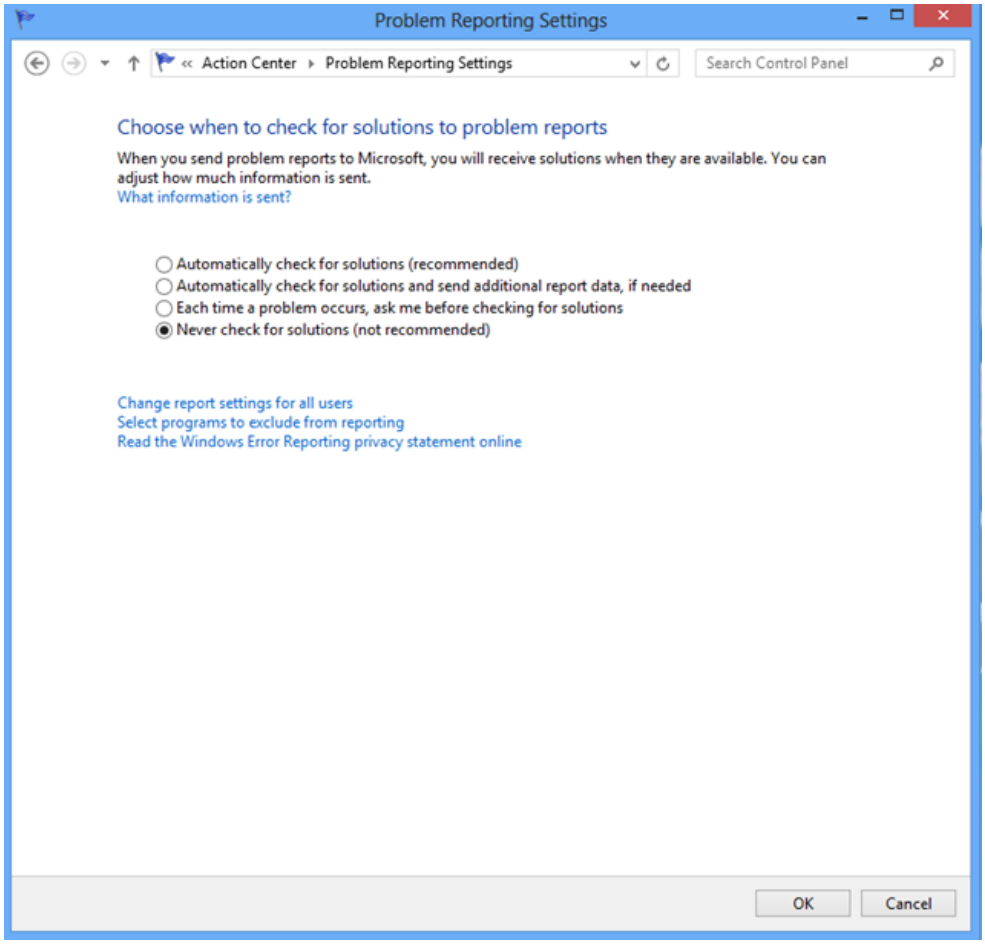


Figure 39 – Inadvertent capture of PAN: Windows 8

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

Addressing Inadvertent Capture of Pan on Windows 10

Disabling System Restore – Windows 10

- Right Click on This PC > Select “Properties”:

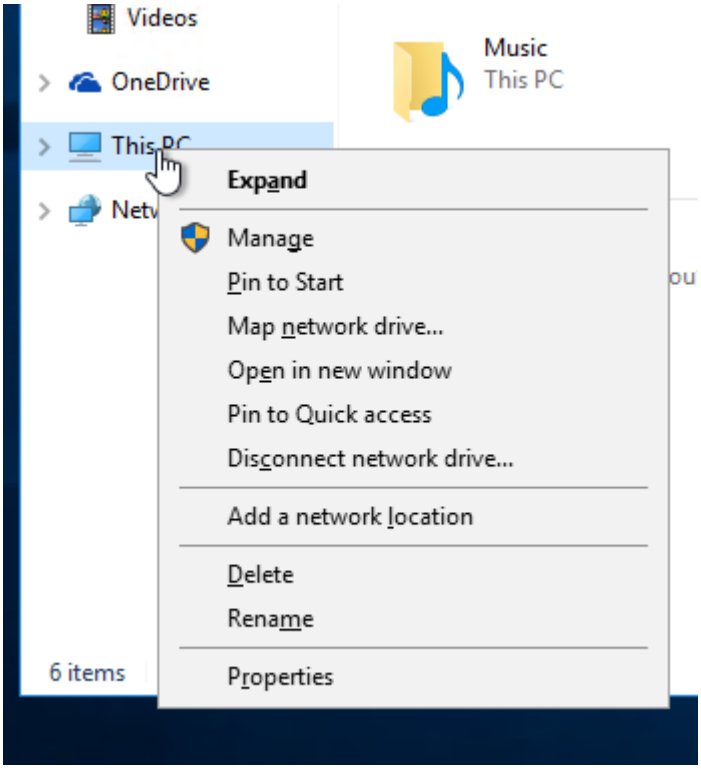


Figure 40 - Inadvertent capture of PAN: Windows 10

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

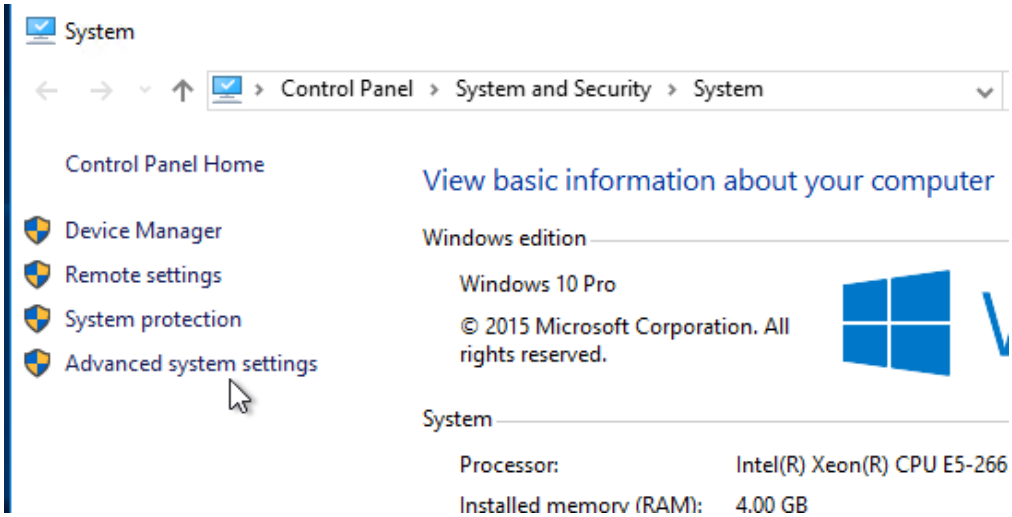


Figure 41 - Inadvertent capture of PAN: Windows 10

- Select “System Protection” tab, the following screen will appear:

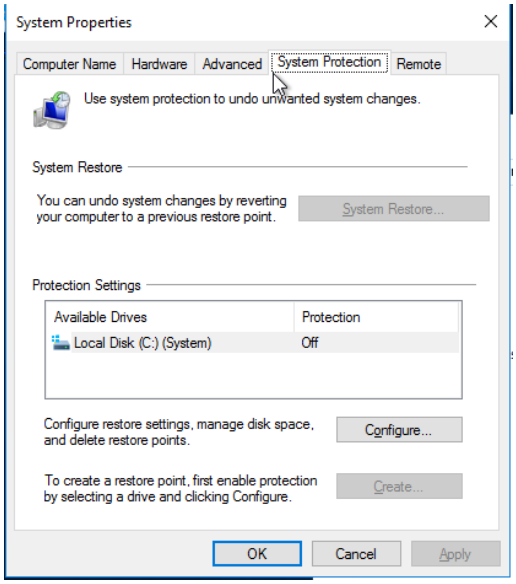


Figure 42 - Inadvertent capture of PAN: Windows 10

- Select Configure, the following screen will appear:

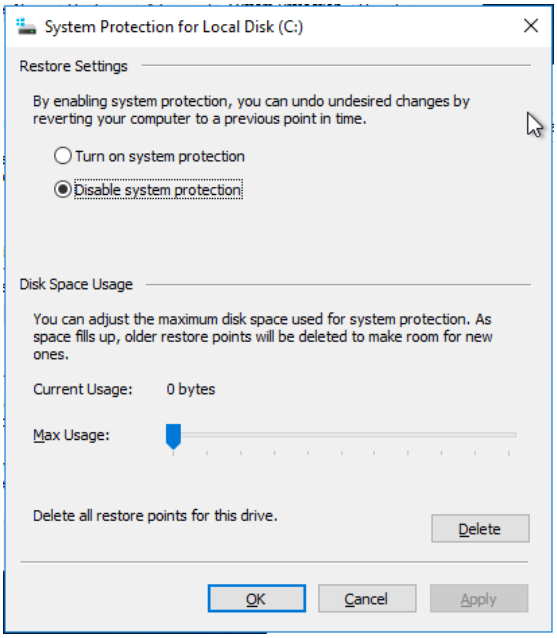


Figure 43 - Inadvertent capture of PAN: Windows 10

- 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 10

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

- From the start menu, 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

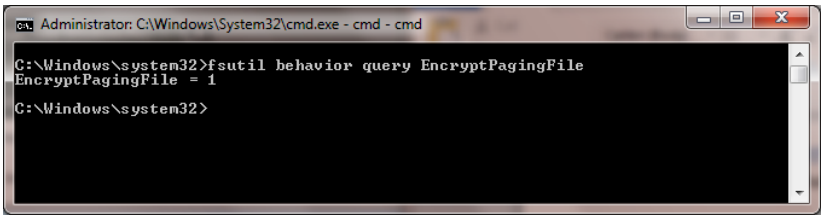


Figure 44 - Inadvertent capture of PAN: Windows 10

- 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

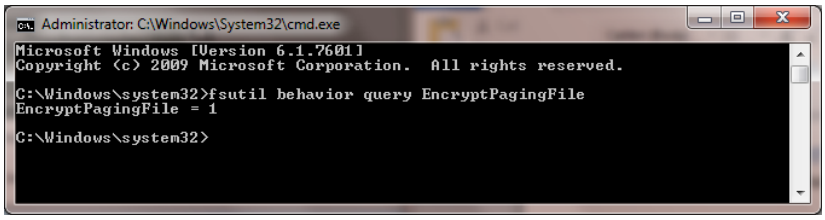


Figure 45 - Inadvertent capture of PAN: Windows 10

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

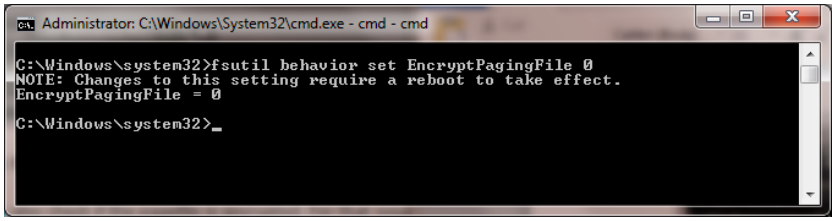


Figure 46 - Inadvertent capture of PAN: Windows 10

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), etc.).

NOTE: Enabling this feature may increase windows shutdown time.

- From the start menu, 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

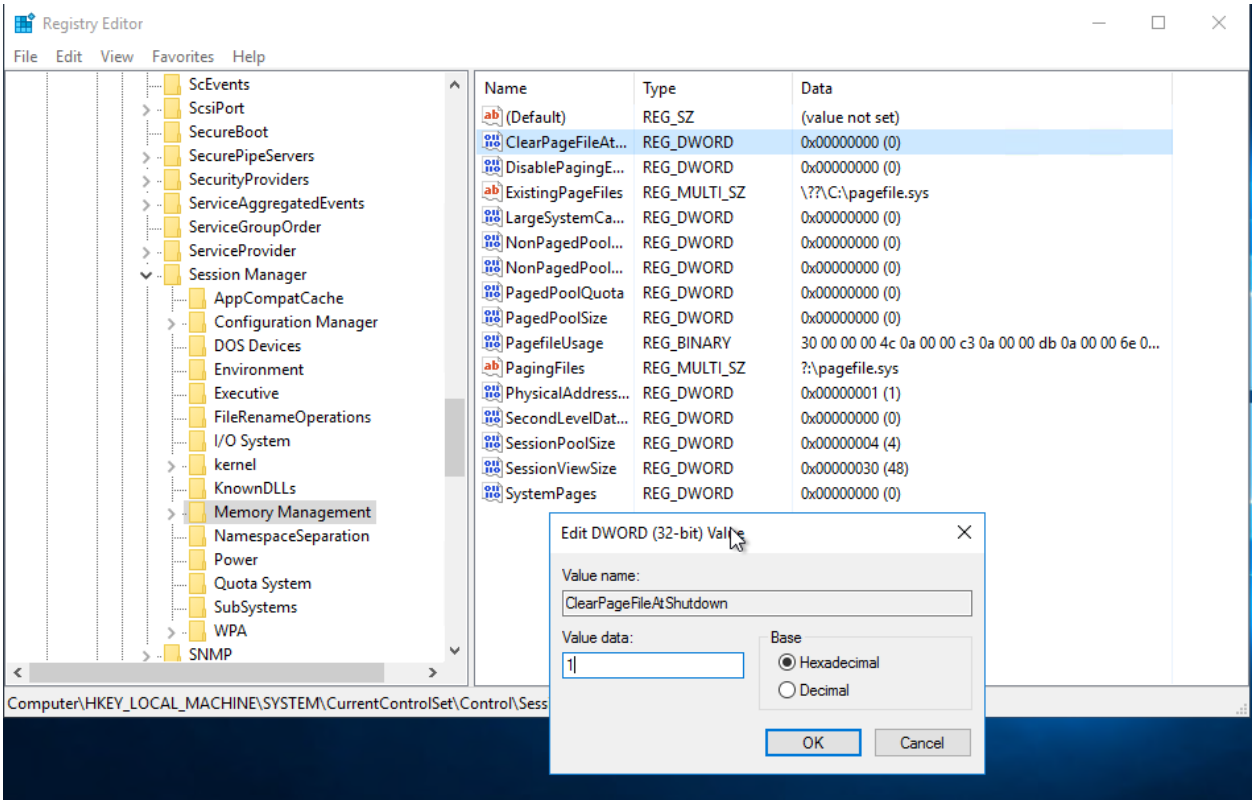


Figure 47 - Inadvertent capture of PAN: Windows 10

- 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 10

- Right Click on This PC > Select “Properties”:

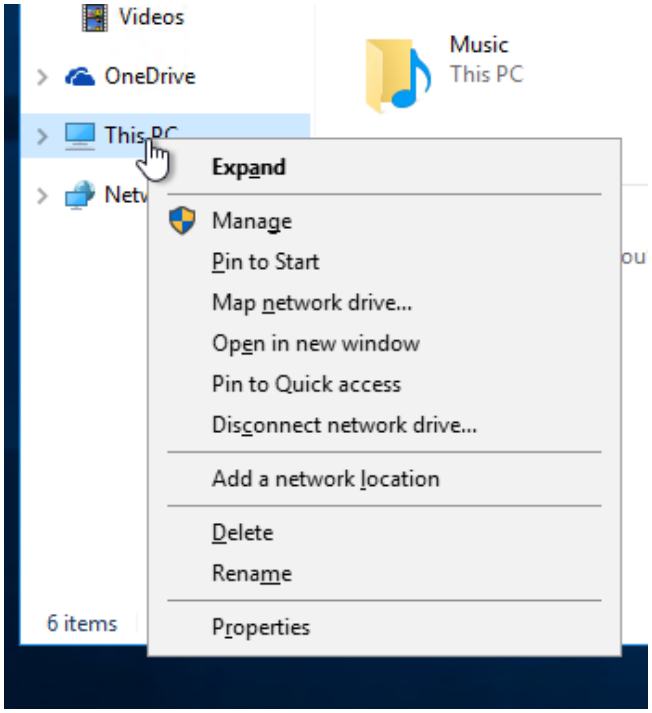


Figure 48 - Inadvertent capture of PAN: Windows 10

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

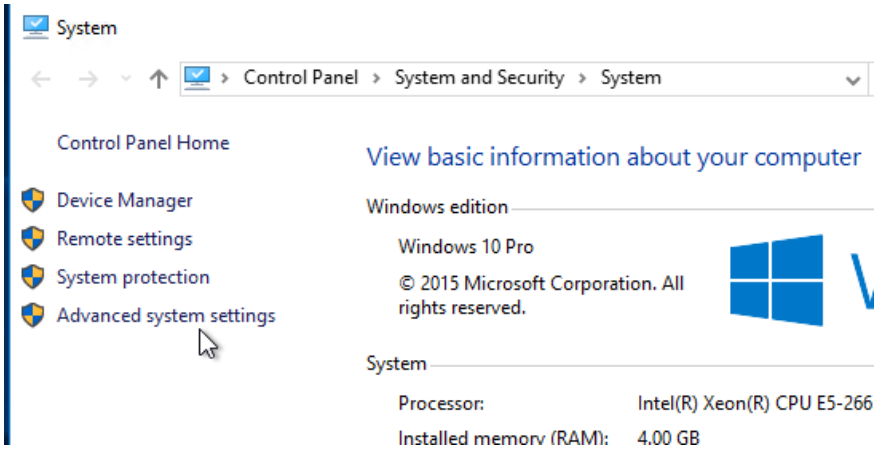


Figure 49 - Inadvertent capture of PAN: Windows 10

- Select the “Advanced” tab:

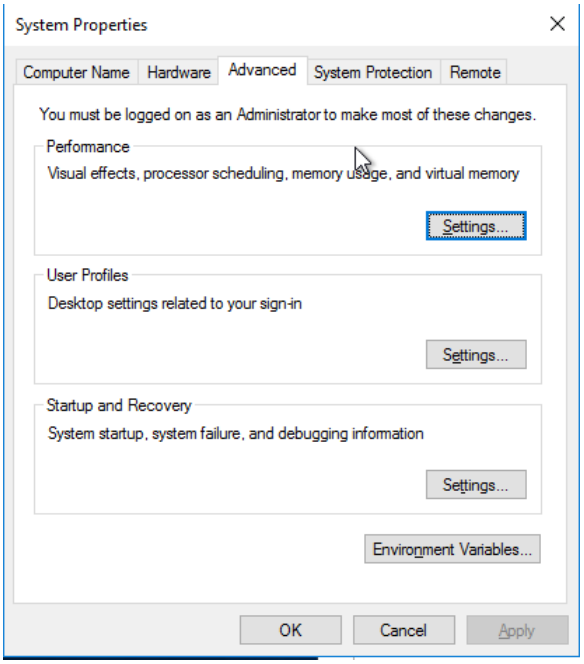


Figure 50 - Inadvertent capture of PAN: Windows 10

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

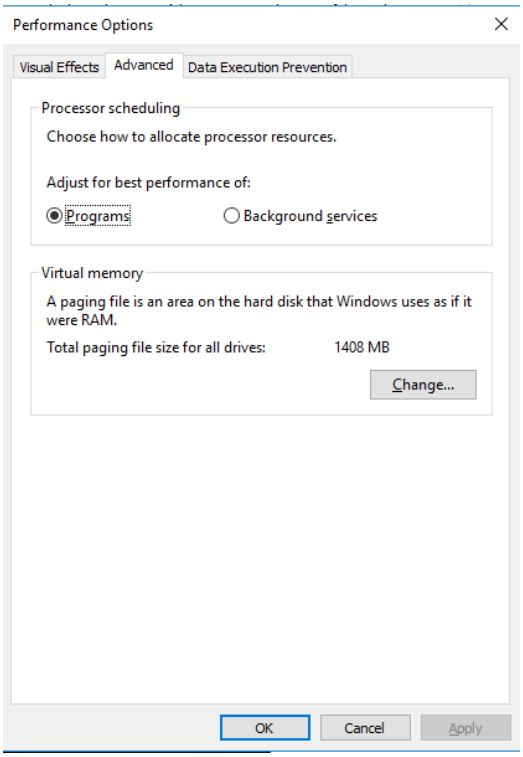


Figure 51 - Inadvertent capture of PAN: Windows 10

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

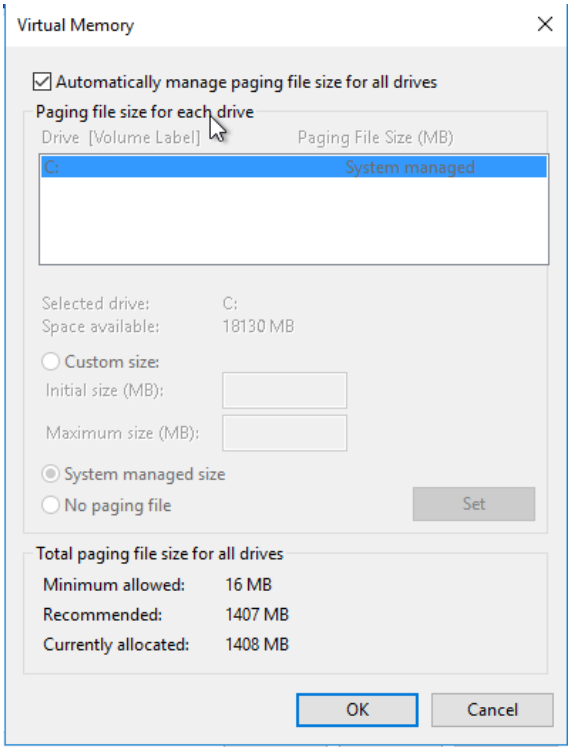


Figure 52 - Inadvertent capture of PAN: Windows 10

- 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 10

- From the start menu, type “control panel”, then enter.
- Open Troubleshooting
- Select ne:

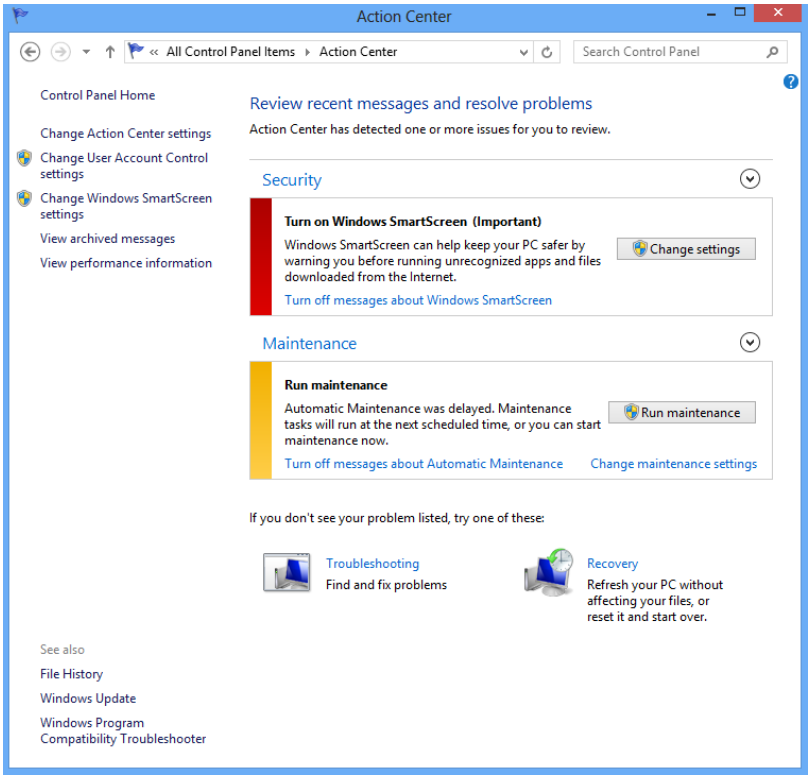


Figure 53 - Inadvertent capture of PAN: Windows 10

- Select “Problem Reporting Settings”:

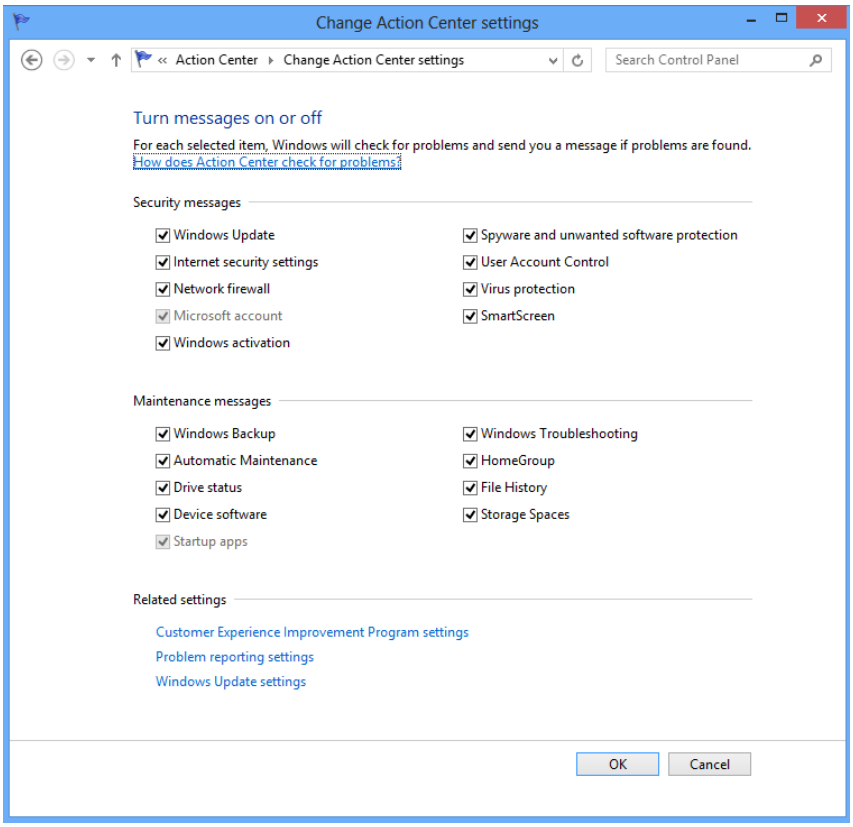


Figure 54 - Inadvertent capture of PAN: Windows 10

- Select “Never Check for Solutions”:

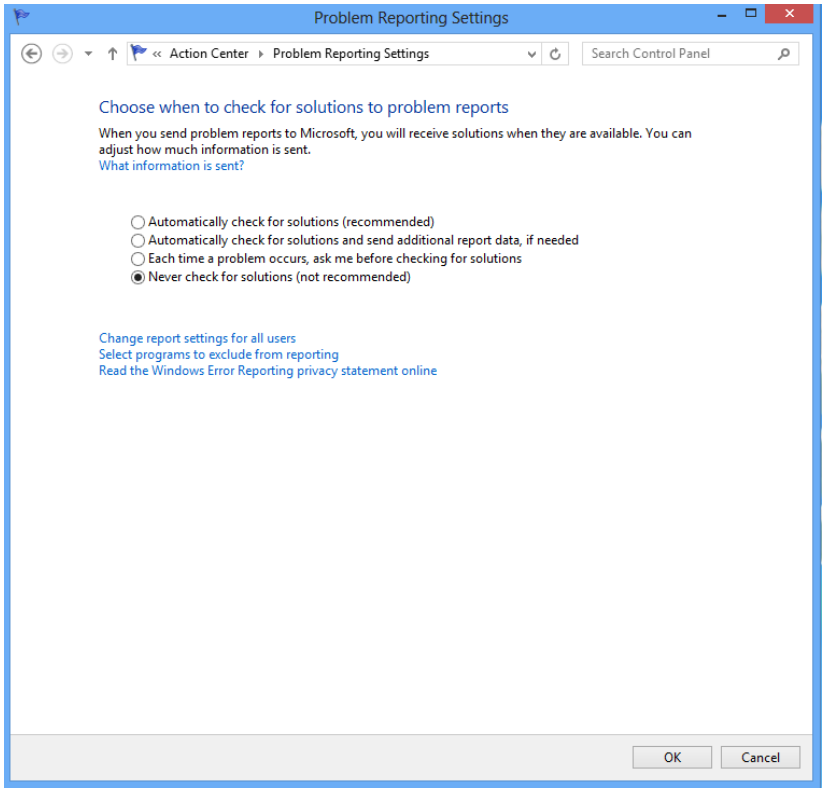


Figure 55 - Inadvertent capture of PAN: Windows 10

Select “OK” twice and then close Action Center.

Addendum – File/Folder Auditing Policy Settings

This information is taken directly from Microsoft

- https://technet.microsoft.com/en-us/library/Cc771070.aspx

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

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:
    - 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**.
    - 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.
    - 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:

- To audit successful events, select the **Successful** check box.
- To stop auditing successful events, clear the **Successful** check box.
- To audit unsuccessful events, select the **Failed** check box.
- To stop auditing unsuccessful events, clear the **Failed** check box.
- 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 – Certificate Validation & Configuration

Certificate Validation

All well-formed certificates contain *Revocation Information* which points to one or more HTTP URI that provide CRL (Certificate Revocation List) and/or OCSP (Online Certificate Status Protocol) resources. The Certificate Authority (CA) specifies the *Revocation Information*: is not controlled by NCR and the URIs specified are not controlled by NCR.

By default, OpenEPS makes use of the Certificate Revocation List URI (CRL), but may also use OCSP if configured to do so. The default operating mode is to use CRL processing because it results in fewer out-bound requests.

OpenEPS performs certificate validation as follows. If any of these steps fail, or cannot be completed, OpenEPS enters into an off-line mode of operation:

1) The Certificate is validated, including the certificate chain
2) The Subject information of the certificate is pinned
3) The Certificate is not revoked

For more information regarding how certificate validation occurs, or how certificates provide security, please see the following resources:

- Certificate path validation algorithm
- An overview of the SSL or TLS handshake
- How SSL and TLS provide identification, authentication, confidentiality and integrity

The final step of certification validation performed by OpenEPS is verification of the revocation information.

Certificate Validation & Off-line Processing

- If the certificate presented to OpenEPS is listed in the revocation list, then OpenEPS enters into an off-line processing mode.
- If OpenEPS cannot access the CRL or OCSP URI, then the certificate presented is deemed as revoked, and OpenEPS enters into an off-line processing mode.

Therefore, it is imperative that your network configuration allow outbound requests to the CRL and/or OCSP services. Should OpenEPS connection attempts to these URIs fail, then the certificate is deemed as revoked and OpenEPS will enter into an off-line processing mode.

CRL Checking on public networks for P2PE

NCR is aware of issues with TLS/SSL session handling for some legacy POS systems. NCR contacted Coalfire, its QSA, to clarify the PCI TLS guidance 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 3.2. These are requirements for NCR's certifications and that means that even though 3DES 168 bit DUKPT encryption is a potential compensating control when dealing with weak TLS/SSL issues, NCR must require that all customers utilize TLS 1.2 and strong ciphers when accessing NCR's Cloud from the Internet.

With regards to customers that are having issues with latency during the certificate validation process, customers can turnoff certificate revocation and use the P2P encrypting solution as a compensating control due to the strong encryption afforded when utilizing 3DES 168 bit DUKPT at the POI Device and still remain complaint with PCI DSS.

However, this statement and guidance is the joint opinion of NCR and Coalfire. The 3DES 168 bit DUKPT encryption in a P2P deployment must be evaluated by your QSA in order for it to be deemed as a compensating control. Additionally, an E2EE solution may also be suitable for this configuration. Consult with your QSA. Your QSA must agree with this opinion and configuration option prior to disabling CRL checking.

Notice:

This configuration is not applicable for any clients running any version of NCR's Enhanced Software Encryption.

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

Certificate Pinning

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

Certificate Pinning

Certificate pinning is the process of both validating the certificate chain and requiring one or more additional elements associated with the certificate to be present and validated to pre-existing conditions and/or expectations. Certificate pinning usually involves maintaining a copy of the certificate for comparison, validating the public key of the, or hashing the certificate or public key and comparing to a known value.

OpenEPS makes use of an alternative method that relies upon certificate chain validation and Subject information pinning.

Configurations for CRL/OCSP Processing

CRL/OCSP URI services for certificate revocation information are provided by the CA (Certificate Authority). These services are not provided or hosted by NCR. Furthermore, the CRL/OCSP URIs are subject to change when

1) NCR deploys new certificates – in advance of expiration of older certificates
2) In the case wherein the CA must revoke one or more certificates due to various circumstances.

Discovery

In order to configure your infrastructure properly, you must discover the CRL/OCSP URIs specified by the Certificates that will be presented by ServerEPS. You do this by using an online tool, such as Qualys® SSL Labs, to inspect the Certificates. See Example CRL/OCSP URI (below) for Connected Payments Certificates known when this document was developed.

Follow these procedures in order to determine which certificates to inspect for revocation information:

Information Gathering Procedures

1) Browse to the OpenEPS directory
    a. C:\Program Files\MicroTrax\OpenEPS
2) Open the **`setup.txt`** file in a text editor
3) Observe and document the URIs (in **bold** below), as associated with the following configuration parameters:
    a. ServerEPSTransactionsHost1=https://**hsptrn1.servereps.com**
    b. ServerEPSTransactionsHost2=https://**hsptrn2.servereps.com**
    c. ServerEPSServicesHost1=https://**hspsvc1.servereps.com**
    d. ServerEPSServicesHost2=https://**hspsvc1.servereps.com**

4) Also document the WWW URI: **https://www.servereps.com**
5) Using an online too, such as Qualys® SSL Labs' "SSL Server Test" (https://www.ssllabs.com/ssltest/), inspect the "Revocation Information" of the Certificate(s) and document the results.
6) Format the data, as appropriate for configuration of your out-bound proxy and/or firewall(s).
7) Deploy the configuration to your test environment
8) Test the configurations and adjust as necessary
9) Deploy the configurations to Production.

Example: Gathering Revocation Information

In order to understand the *Revocation Information* for a given certificate, you must make use of one or more tools that can download the certificate and extract the information for inspection. Qualys® SSL Labs provides an online service to perform certificate inspection.

1) Navigate to this URI: https://www.ssllabs.com/ssltest/
2) One at a time, enter the documented host names, as gathered above.

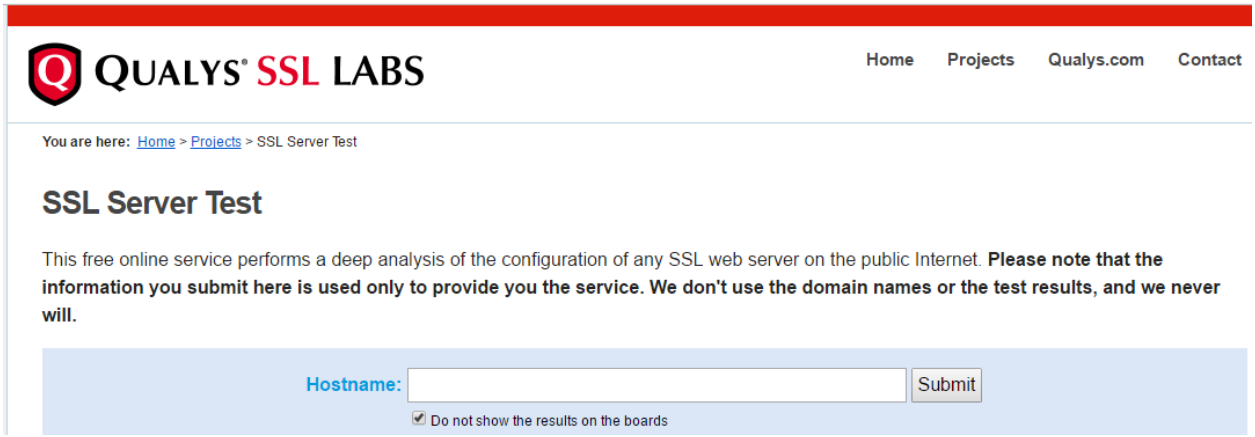


Figure 56 – Qualys® SSL Labs - Query

3) Be sure to click on the "Do not show the results on the boards" check-box.
4) Click the "submit" button and observe the results, as follows.

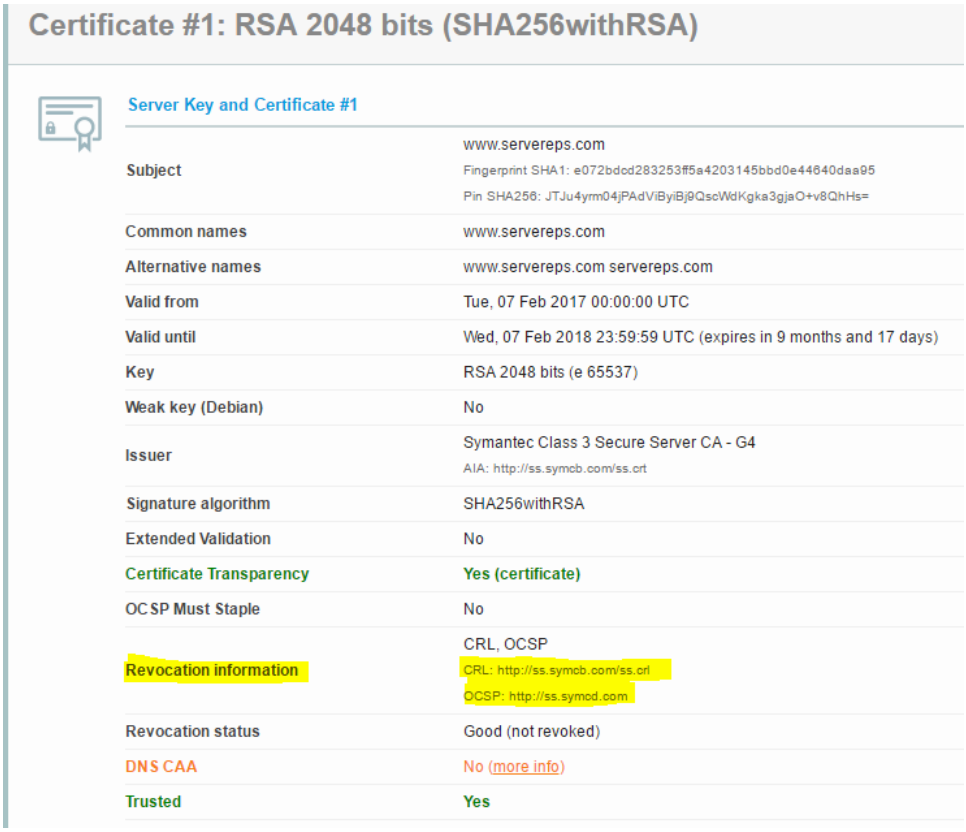


Figure 57 – Qualys® SSL Labs – Results

5) For each certificate inspected, document the *Revocation Information*, show above.

Firewall / Proxy Configuration Guidance

Consult with your Firewall and/or Proxy vendors' support desk before making changes to your firewall configuration.

As will all information system changes, follow your change management policy and risk management policies when making these changes.

Use the following guidelines, while consulting with your firewall vendor, for developing one or more regular expressions suitable Certificate Revocation Information configuration.

NOTICE

- Do not use these examples as-is without thorough testing for
  - applicability and suitability in your infrastructure
  - configuration compatibility with your infrastructure
- You must always test all firewall configurations before moving changes into production

- You must validate your configurations with your firewall vendor or other appropriate Subject Matter Experts
- You must employ best practices change management procedures

Example CRL/OCSP URI

Your documented CRL/OCSP URIs may look similar to the following:

- ss.symcd.com
- ss.symcb.com/ss.crl

Example Regular Expressions

Depending on the available configuration of your firewall and/or proxy servers, you may be able to condense these URIs into one or more regular expressions, as follows:

- ^.+\.symc[db]\.com

Otherwise, you may need to create distinct URI expressions, as follows.

- *.symcd.com
- *.symcb.com

The particular configurations required for your firewall are manufacturer and device dependent. Refer to your user firewall user manual for specific configuration instructions.

Summary

CRL/OCSP processing is a necessary function of OpenEPS. If your out-bound (intranet to internet) configurations prevent out-bound connections on port 80 for the HTTP protocol, then OpenEPS will enter into an off-line mode of processing: transactions/sales will not be processed. You must enable port 80 (protocol HTTP) out-bound processing for the URIs provided in the certificates presented by ServerEPS, as configured in your *setup.txt* file and for www.servereps.com.

Addendum – TLS and FIPS 140-2

FIPS 140-2

OpenEPS does not rely upon the underlying Operating System for any encryption services it might provide during its normal operation. OpenEPS is specifically designed to provide encryption, hashing and digital signature functions for data at rest, data in use and data in motion, autonomously.

OpenEPS uses only TLS v1.2 cipher suites that are FIPS 140-2 approved. These services are built into OpenEPS using WolfCrypt (by WolfSSL). More information regarding WolfCrypt, and FIPS 140-2 may be found at the following references:

- FIPS 140-2
  - http://csrc.nist.gov/groups/STM/cmvp/standards.html
- WolfSSL
  - https://www.wolfssl.com/wolfSSL/Home.html
- **WolfCrypt** FIPS 140-2 Level 1 Certificate #2425:
  - http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm#2425

TLS v1.2

To lessen the dependency on the Operating System for PCI compliancy, NCR has embedded OpenEPS with TLS 1.2 cipher suites and session management, to include ECC, DH, RSA, SHA2, and AES. NCR recommends that merchants update to the latest version of OpenEPS in order to provide the strongest level of session encryption, and to enable them to be prepared for PCI 3.2 TLS v1.2 compliancy.

OpenEPS supports the following cipher suites, regardless of the Operating System upon which it is installed:

TLS_ECHD_ECDSA_WITH_AES_128_GCM_SHA256
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
TLS_DH_RSA_WITH_AES_128_GCM_SHA256
TLS_DH_RSA_WITH_AES_256_GCM_SHA384
TLS_ECHDE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECHDE_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECHDE_RSA_WITH_AES_128_GCM_SHA256
TLS_ECHDE_RSA_WITH_AES_256_GCM_SHA384
TLS_ECHD_ECDSA_WITH_AES_256_GCM_SHA384

TLS_ECHD_RSA_WITH_AES_128_GCM_SHA256

TLS_ECHD_RSA_WITH_AES_256_GCM_SHA384

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_128_GCM_SHA256

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_AES_256_GCM_SHA384

Addendum – Firewall Configuration

Keeping the Internet Out

OpenEPS requires the following:

1) An internet connection from each point of sale lane to Connected Payments
2) A properly configured firewall

The goal of this document is to provide basic instructions on how to configure your network firewall for a CDE to allow specific and limited outbound connections while eliminating undesired incoming connections from the internet.

Outbound and Inbound Connections

An outbound communication is one that originates within the network and connects to a provider outside the network. Inbound or incoming connections originate outside the network with a target host computer that is inside the network.

Inbound connections generally represent the most common threat to network security; hackers on the internet can use scanning software to locate mis-configured or unprotected open ports and use these ports to bypass security. Most firewall hardware and software limit or eliminate inbound traffic as part of their default settings; this is the main reason that use of firewalls is required by PCI regulations.

Outbound connections pose less of a security risk as long as the software initializing the connection is known and trusted. The security of even known and trusted software can be augmented with the proper use of firewalls and Access Control Lists (ACLs), while at the same time ensuring outbound access from unknown or untrusted software can be completely denied.

The OpenEPS Direct payments solution has been designed with network security in mind. As such it does not require any inbound connections; when a lane starts up, that lane initiates an outbound connection to the payments host – no incoming connection is required.

With that in mind, it is a simple matter to configure your firewall to allow the OpenEPS Direct software to connect to just the payments host, and to prevent all other network traffic either inbound or outbound.

Firewalls

A firewall can come in the form of software loaded onto a computer, or as a separate piece of hardware that connections are routed through. PCI requires the use of a firewall in the payments environment, and firewalls themselves are easy to obtain by searching online for free software firewalls or locating a hardware firewall at your local electronics store.

The benefit of software firewalls is that they are inexpensive and often free. Zone Alarm is an excellent example of free firewall software that defaults to denying access from any other computer and can be configured to allow only specific programs to connect out to the internet. Software firewalls do tend to require an installation and configuration on each system to be protected; this can mean installing a software firewall at each POS lane.

If you are running a Windows XP system, don't rely on the built-in Windows XP firewall as it doesn't block or control outgoing connections.

Hardware firewalls come in a variety of grades ranging from protecting a small home network to a large company network. Linksys, Belkin and Netgear all offer low cost ($50 to $60) consumer-grade router/firewalls for small networks that provide the ability to restrict access based on network IP addresses. Somewhat more expensive business grade firewalls will offer more options for limiting the connectivity by start and destination IP address and by port number.

One significant advantage a hardware firewall has over software firewalls is a central location for management. While software firewalls generally require setup on each machine they are installed on, a hardware firewall can be configured to allow or deny access to a range of IP addresses, making configuration of rules for a large number of POS lanes much simpler.

Keep in mind that PCI requires that payments hardware be kept in a safe location. As a part of the payments network, your firewall hardware should be placed in a secure location, such as a locked server closet.

Additional Safety Measures

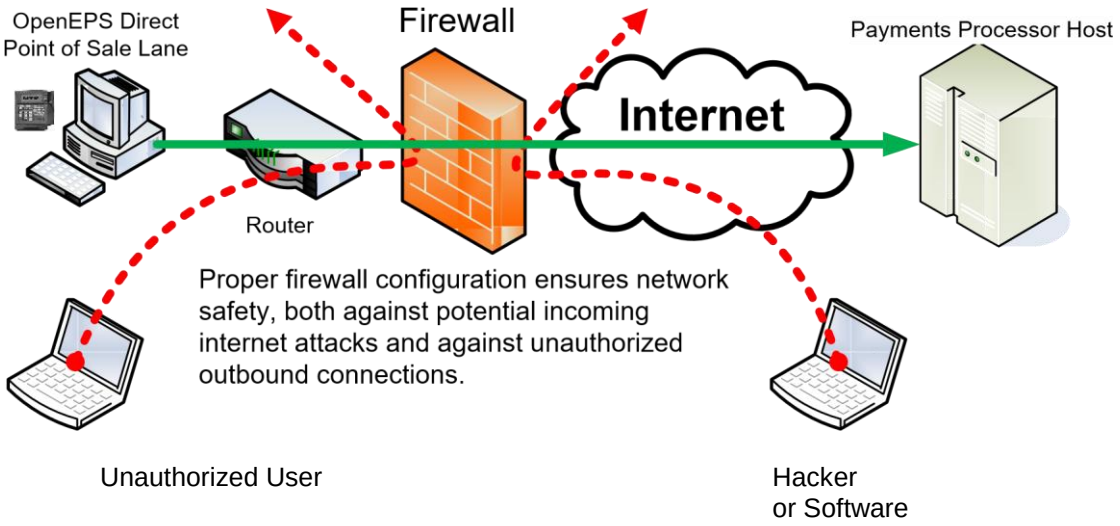
In addition to a firewall, use of Change detection and Configuration Control software, such products listed below.

- Lumension - https://www.lumension.com/Company/About-Us.aspx
- McAfee - https://www.mcafee.com/in/solutions/retail.aspx

- TripWire - https://www.tripwire.com/solutions/solutions-by-industry/retail-and-hospitality/tripwire-solutions-for-retail-security-register/
- Cimcor - http://www.cimcor.com/cimtrak/

Connections: Trusted Software to Trusted Sites

Firewalls keep a network safe by denying access. To properly configure a firewall it is important to focus on what software you want to use and to what host(s) that software needs to connect. The more information you have about your software and connections, the more specific you can make your firewall rules, and the more secure your network becomes.



As you can see from the diagram above, it is possible to configure a firewall to deny unauthorized outbound connections from within the network, as well as unwanted connections from the internet while allowing the required connection from OpenEPS to the payment host provider.

The connection details that follow should allow you to configure your firewall to maximize network protection. Precisely what options you have for firewall connection rules is dependent on what your firewall allows, but the more specific you can make the rules the better the network is protected.

POS Lane Connections

The OpenEPS Direct solution resides at each POS lane, and connects to payments hosts with fixed DNS names such as Trn1.ServerEPS.com, Trn2.ServerEPS.com, Svc1.ServerEPS.com, and Svc2.ServerEPS.com. These connections occur on port 443 as shown on the chart below.

Firewall Configuration Information

Host DNS Name	Service	Host Port
Trn1.ServerEPS.com through Trn6.ServerEPS.com	Primary and Backup Transaction Processing	443
Svc1.ServerEPS.com through Svc3.ServerEPS.com	Primary Configuration Download	443

IP Address Ranges	Location	Host Port
4.79.143.162 – 4.79.143.174	Data Center 1	443
208.80.28.162 – 208.80.28.190	Data Center 2	443

Additional servers are added from time to time and the IP address of existing servers may change; therefore, it is recommended that a DNS server be used instead of utilizing the IP address. The IP addresses are included for completeness.

Using this information, it is possible to configure your network firewall to allow the OpenEPS Direct software to connect out to only the payments host addresses listed and prevent any other connection from being established.

Report Service, Web Site Access

In addition to the POS lanes, it is likely necessary that at least one PC at the store will need to be able to log into the online Report Service for the OpenEPS Direct product. The report service is available at www.servereps.com and communicates on port 443.

Report Service Host DNS Name	IP Address	Host Port
www.ServerEPS.com	4.79.143.167	443
www.ServerEPS.com	4.79.143.167	80

When a user signs on to www.servereps.com using their internet browser, the initial connection is generally established on port 80 (http protocol) and then switches over to secure port 443 (https protocol) automatically as the session begins.

If it is desirable to entirely block outbound traffic on port 80, then a simple desktop shortcut should be included that points to https://www.servereps.com to initiate the connection on secure port 443 to begin with.

Secure Desktop Shortcut Link
https://www.servereps.com

Similar to the POS lanes, this connection can be limited to only the computers that require it and the connection can be limited to only the required site.

Firewall Configuration Example

This example of a hardware firewall uses a Linksys model router. This router is relatively inexpensive and uses a Graphical interface to configure. This router allows centralized security management that features security policies that can be set to pertain to a range of IP addresses, such as are used for POS lanes.

The router first needs to be installed on the network between the internet and the POS lanes so that connectivity to the internet must pass through the router. After the router is installed and properly configured to allow basic connectivity, you may use the Access Restrictions Tab to configure your security policy.

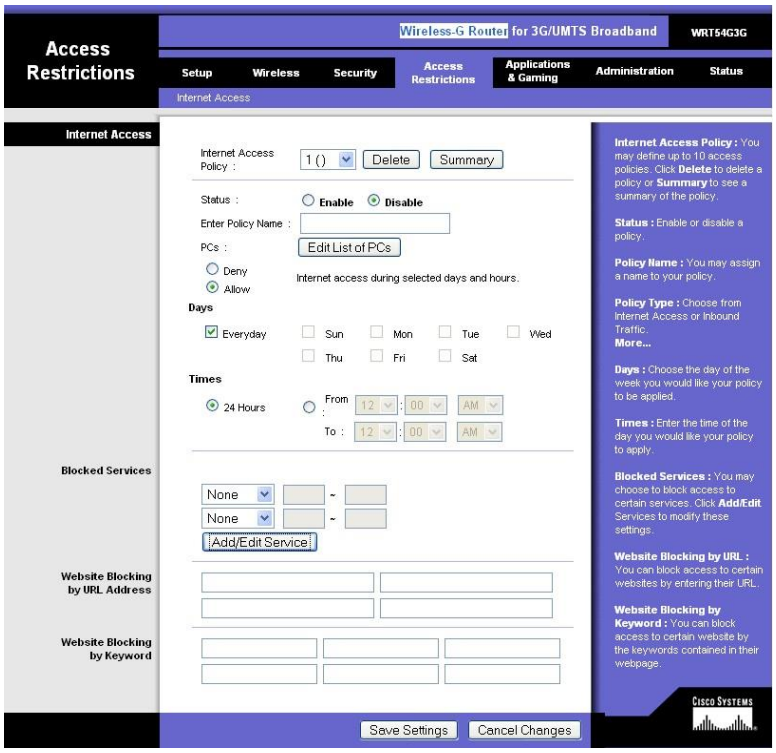


Figure 58 - Firewall Access Restrictions

Click the “Edit List of PCs” button to define the list of computers to which this policy will apply.

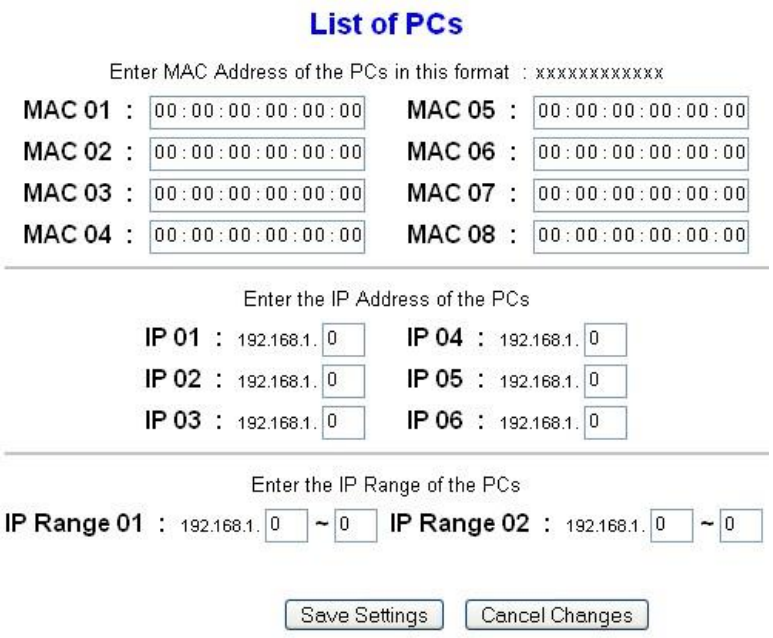


Figure 59 - Firewall List of PCs

To select a range of PCs, use the IP Range options at the bottom. You may use this option to apply the policy to all your Point of Sale lanes, for example, by specifying the lowest IP a POS lane is assigned, through the highest IP address your POS lanes are assigned. Be sure to select Save after making changes on this screen.

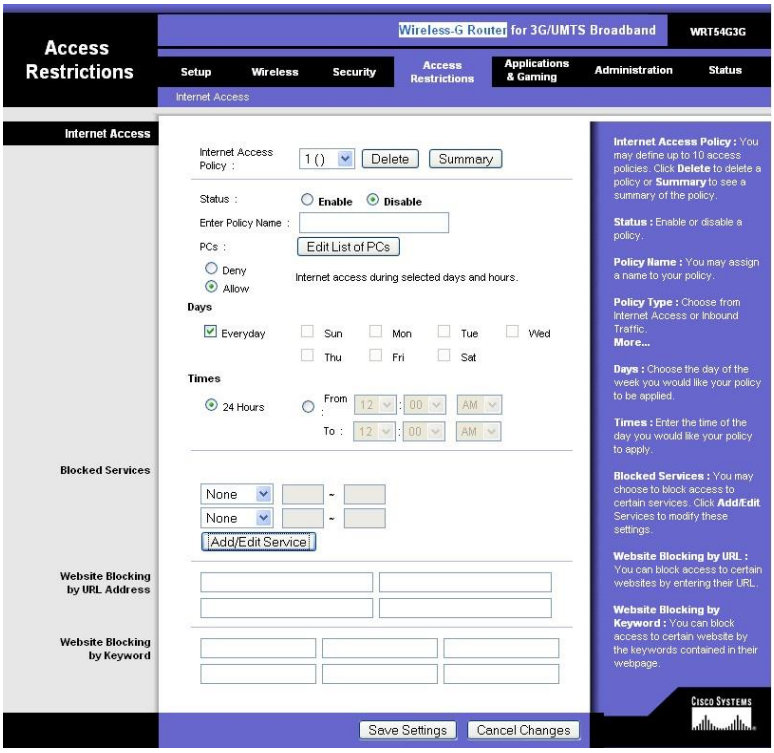


Figure 60 - Firewall Access Restrictions

Returning to the main Access Restrictions Tab, the next option to configure are the blocked services. To enable configuration of Blocked Services, select the “Allow” option under “PCs:”.

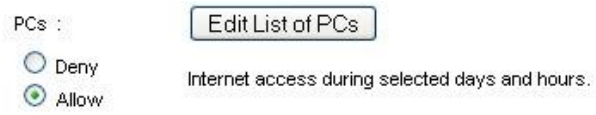


Figure 61 - Allow PCs

On this router it is only possible to block up to two ranges of services per security policy, but two ranges should be all that is needed to block all ports from the POS lanes except port 443 that OpenEPS Direct requires to connect out to the payments host servers.

You can create two new services that will include all the port you need to block by selecting the “Add/Edit a Service’ button.

Figure 62 - Firewall Service Configuration

On the screen above, enter a new service name and use the Protocol dropdown to select TCP & UDP. Enter a range of 0 to 442 and click Add. Enter a second Service Name and enter the ports 444 to 65535. Click Add to save.

Returning to the Access Restrictions Tab, use the first service selection dropdown in the Blocked Services section to select the first port range name you created. Use the second dropdown to select the second range.

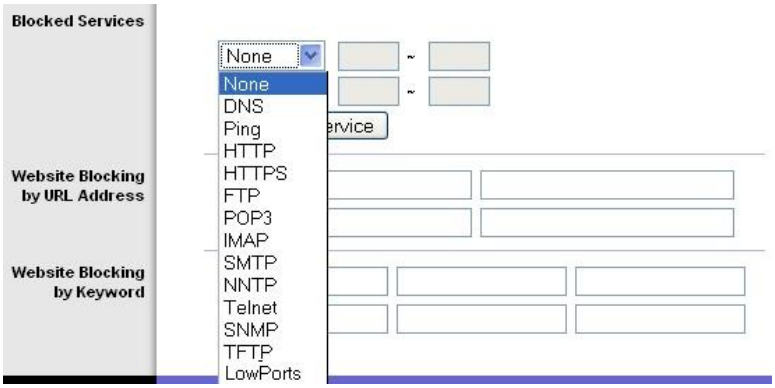


Figure 63 - Firewall Range Selection

The ranges displayed should show the ranges you entered, 0 to 442 and 444 to 65535. This combination of port filters leaves only port 443, secure HTTP open for connection for the range of POS lane IPs you entered above. Be sure to select Save to keep the changes you have made.

This same policy can be used to restrict outbound connectivity to a non-POS lane computer, such as the PC on which users will review the reports available at www.servereps.com. The above policy will only allow that PC to connect on port 443, and not port 80 which is typically used for internet connectivity. In this event, be sure to set up the shortcut to take the user directly to the proper site as detailed in the Report Service, Web Site Access section.

Addendum – Definitions

Definitions are provided here for your convenience for the terms used in this document. For the most up-to-date official glossary of PCI SSC terms, refer to the following official PCI references:

- PCI Glossary of Terms, Abbreviations, and Acronyms
- Small Merchant Glossary of Payment and Information Security Terms

Term	*Definition*
3DES	See Triple DES.
Cipher	See Cryptographic Algorithm.
Ciphertext	Encrypted text as derived from plaintext, when employing an encryption or cryptographic, algorithm.
Cardholder Data	At a minimum, cardholder data consists of the full PAN. Cardholder data may also appear in the form of the full PAN plus any of the following: cardholder name, expiration date and/or service code.
Cardholder Data Environment	The people, processes and technology that store, process, or transmit cardholder data or sensitive authentication data.
CHD	Acronym for Cardholder Data.
CDE	Acronym for Cardholder Data Environment
Connected Payments	The NCR, PCI-DSS certified application responsible for payment processing. Also known as ServerEPS. OpenEPS establishes a secure session to Connected Payments for various reasons including but not limited to: self-referential integrity checking and all supported types of payment processing.
CRL	Certificate Revocation List. A list of revoked certificates as defined in RFC 5280, "*Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*." CRL processing is supported using various operational protocols, such as LDAP, HTTP, FTP and X.500. See *OCSP*
Cryptographic Algorithm	A data encrypting scheme. A well-defined and repeatable process used to convert plaintext into ciphertext in such a way that the ciphertext can be accessed without revealing the plaintext. Examples are: AES-256, DUKPT, RSA.
Decryption	The process of rendering a ciphertext into the plaintext from which the ciphertext was derived.
DEK	Data Encrypting Key. The key used by a data encrypting scheme to encrypt sensitive data.
DES	Data Encryption Standard. An encryption standard developed by IBM in the early 1970s, used in the Triple Data Encryption Algorithm (TDEA).

Term	*Definition*
DUKPT	Derived Unique Key Per Transaction. A Key management scheme in which a unique key is derived from a base or fixed key for every transaction. The cryptographic algorithm is designed in such a way that if the derived key is compromised, prior and future keys cannot easily be determined. Defined in ANSI X9.24 part 1.
E2EE	End to End Encryption. A solution that defines a message path and a single encryption protocol and key, as well as the systems used to implement the transmission of encrypted messages between two primary communicating partners. In E2EE, at no point between the two primary communicating partners can the encrypted message be changed, inspected or decrypted by any partner or system in the message path.
Encryption	A reversible process of encoding plaintext into ciphertext that permits only authorized entities to access the plaintext. The plaintext is rendered into ciphertext using an encryption algorithm that can be converted back into plaintext only if decrypted.
ESE	Enhanced Software Encryption. A method wherein the POI Device uses a salt provided by OpenEPS and a KEK provided by Connected Payments to encrypt CHD and the DEK.
Hash	A non-reversible process of encoding plaintext into message digest. Once the digest is created, it cannot be reversed to produce the original text. Some hashing algorithms have been compromised through collision attacks wherein two different or similar plaintexts have been demonstrated to produce the same hash or message digest. See NIST SP 800-106, FIPS PUB 180-4.
KEK	Key Encrypting Key. The key used by a data encrypting implementation to encrypt a DEK (Data Encrypting Key). For OpenEPS, the KEK is a 4096 bit, RSA public key.
OCSP	Online Certificate Status Protocol. A protocol used to determine the status of a digital certificate without requiring Certificate Revocation Lists (CRL), as defined in RFC 6960. OCSP is supported using the HTTP protocol.
P2PE	Point to Point Encryption. A solution that defines a message path and one or more encryption protocols and keys, as well as the systems used to implement the transmission of encrypted messages between two primary communicating partners. In P2PE, the encryption algorithm and keys change between the various communicating partners so that each partner along the message path may inspect or modify the encrypted message as necessary before transmitting the message to the next partner in the message path. The P2PE acronym is never used in this document to infer a *PCI P2PE Validated Solution* or component.
P2PE Solution	A combination of secure devices, applications, and processes that encrypt cardholder data from a PCI-approved point-of-interaction (POI Device) device through to decryption, assessed in accordance with PCI's P2PE standard and included on PCI's list of Validated P2PE Solutions
PAN	Primary Account number. Also referred to as "account number." Unique payment card number (typically for credit or debit cards) that identifies the issuer and the particular cardholder account.

Term	*Definition*
PA-DSS	The Payment Application Data Security Standard. Previously known as the Payment Application Best Practices (PABP). It is the security standard created by the Payment Card Industry Security Standards Council (PCI SSC) for payment applications.
PA-QSA	Payment Application Qualified Security Assessor Acronym for "Payment Application Qualified Security Assessor." PA-QSAs are qualified by PCI SSC to assess payment applications against the PA-DSS.
PII	Personally identifiable information. Any data that could potentially identify a specific individual. Information that can be used to distinguish one person from another.
PIN Entry Device	An electronic device used in a debit, credit or smart card-based transaction to accept and encrypt the cardholder's personal identification number (PIN). See also *POI Device* and *POI Device.*
PIN	Personal Identification Number.
POI Device	The *Point of Interaction*, or *PIN Entry Device.* The initial point where data is read from a card. An electronic transaction-acceptance product, a POI Device consists of hardware and software and is hosted in acceptance equipment to enable a cardholder to perform a card transaction. The POI Device may be attended or unattended. POI Device transactions are typically integrated circuit (chip) and/or magnetic-stripe card-based payment transactions. See also *POI Device.*
PINPAD	Point of Interaction or *PIN Entry Device*. See also *POI Device*.
POS	Point of Sale. The hardware and/or software used to process payment card transactions at merchant locations. It is used by merchant employees to tabulate goods sold, accept and store cash and print receipts.
PTS	PIN Transaction Security. A set of modular evaluation requirements managed by PCI Security Standards Council for PIN acceptance POI Device terminals.
POI Device	PIN Transaction Security (PTS) Point of Interaction (POI Device) used for capturing payment card data. Also known as a *POI Device*, *PIN Entry Device* or *PINPAD*. The Council, via PCI Recognized Laboratories, validates the conformance of PTS devices to the PCI PTS standard and provides a list of approved devices. See https://www.pcisecuritystandards.org/assessors_and_solutions/pin_transaction_devices
RSA	An asymmetric-key encryption algorithm. Named after Rivest, Shamir, and Adleman.
SAD	*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.
ServerEPS	See Connected Payments.
TDEA	See Triple DES.

Term	*Definition*
Triple DES	Triple Data Encryption Algorithm (TDEA). A symmetric-key block cipher which applies the Data Encryption Standard (DES) three times to each data block. Also known as Triple Data Encryption Algorithm (TDEA). See also: ANSI X9.52, NIST SP 800-67, FIPS PUB 46-3.

Addendum – Table of Figures

Figure 1 – Generic Network Dataflow Diagram 19

Figure 2 - P2P Data Flow Diagram 20
Figure 3 - E2EE Data Flow Diagram 22
Figure 4 – ESE Data Flow Diagram 24
Figure 5 - Manual PAN/CVV Data Entry Flow 26
Figure 6 - OpenEPS Directory 32
Figure 7 - ServerEPS Screen-shot for OpenEPS Key re-generation 36
Figure 8 – Passwords, Local Group Policy Editor 39
Figure 9 - Passwords, Local Group Policy Editor 40
Figure 10 - Screen Saver Configuration 41
Figure 11 – Inadvertent capture of PAN: Windows 7 68
Figure 12 – Inadvertent capture of PAN: Windows 7 69
Figure 13 – Inadvertent capture of PAN: Windows 7 70
Figure 14 – Inadvertent capture of PAN: Windows 7 70
Figure 15 – Inadvertent capture of PAN: Windows 7 70
Figure 16 – Inadvertent capture of PAN: Windows 7 71
Figure 17 – Inadvertent capture of PAN: Windows 7 71
Figure 18 – Inadvertent capture of PAN: Windows 7 72
Figure 19 – Inadvertent capture of PAN: Windows 7 73
Figure 20 – Inadvertent capture of PAN: Windows 7 74
Figure 21 – Inadvertent capture of PAN: Windows 7 75
Figure 22 – Inadvertent capture of PAN: Windows 7 75
Figure 23 – Inadvertent capture of PAN: Windows 7 76
Figure 24 – Inadvertent capture of PAN: Windows 8 77
Figure 25 – Inadvertent capture of PAN: Windows 8 77
Figure 26 – Inadvertent capture of PAN: Windows 8 78
Figure 27 – Inadvertent capture of PAN: Windows 8 79
Figure 28 – Inadvertent capture of PAN: Windows 8 80
Figure 29 – Inadvertent capture of PAN: Windows 8 80
Figure 30 – Inadvertent capture of PAN: Windows 8 80
Figure 31 – Inadvertent capture of PAN: Windows 8 81
Figure 32 – Inadvertent capture of PAN: Windows 8 82
Figure 33 – Inadvertent capture of PAN: Windows 8 82
Figure 34 – Inadvertent capture of PAN: Windows 8 83
Figure 35 – Inadvertent capture of PAN: Windows 8 84
Figure 36 – Inadvertent capture of PAN: Windows 8 85
Figure 37 – Inadvertent capture of PAN: Windows 8 86
Figure 38 – Inadvertent capture of PAN: Windows 8 87
Figure 39 – Inadvertent capture of PAN: Windows 8 88
Figure 40 - Inadvertent capture of PAN: Windows 10 89
Figure 41 - Inadvertent capture of PAN: Windows 10 90
Figure 42 - Inadvertent capture of PAN: Windows 10 90

Figure 43 - Inadvertent capture of PAN: Windows 10 91
Figure 44 - Inadvertent capture of PAN: Windows 10 91
Figure 45 - Inadvertent capture of PAN: Windows 10 92
Figure 46 - Inadvertent capture of PAN: Windows 10 92
Figure 47 - Inadvertent capture of PAN: Windows 10 93
Figure 48 - Inadvertent capture of PAN: Windows 10 94
Figure 49 - Inadvertent capture of PAN: Windows 10 94
Figure 50 - Inadvertent capture of PAN: Windows 10 95
Figure 51 - Inadvertent capture of PAN: Windows 10 95
Figure 52 - Inadvertent capture of PAN: Windows 10 96
Figure 53 - Inadvertent capture of PAN: Windows 10 97
Figure 54 - Inadvertent capture of PAN: Windows 10 98
Figure 55 - Inadvertent capture of PAN: Windows 10 98
Figure 56 – Qualys® SSL Labs - Query 106
Figure 57 – Qualys® SSL Labs – Results 107
Figure 58 - Firewall Access Restrictions 116
Figure 59 - Firewall List of PCs 117
Figure 60 - Firewall Access Restrictions 117
Figure 61 - Allow PCs 118
Figure 62 - Firewall Service Configuration 118
Figure 63 - Firewall Range Selection 119