WAY4™ PA-DSS Implementation Guide

Contents

Introduction

This document covers requirements and settings in WAY4™ modules necessary for compliance with the Payment Card Industry Data Security Standard (PCI DSS).

This document is not a full guide to WAY4 installation and setup, however these requirements must be met for compliance with PCI DSS v. 3.2.

Compliance with security requirements minimises the potential for compromises of sensitive card data such as card track contents, card verification values (CAV2, CID, CVC2, CVV2), PIN code and PIN block data and accordingly, aids in eliminating the risk of fraud using payment card data.

Compliance with Visa 3-D Secure and MasterCard SecureCode requirements for WAY4 components implementing 3-D Secure functionality is separate from PA-DSS validation and is the subject of separate testing with Visa and Mastercard, respectively.

This document is intended for system administrators (bank and processing centre employees) responsible for creating and supporting the network infrastructure for WAY4 applications, administering WAY4 applications and performing various administrative security-related functions.

This document is based on the following sources:

- "Payment Card Industry (PCI) Data Security Standard" Version 3.2 April 2016".
- "Payment Card Industry (PCI) Data Security Standard. Glossary, Abbreviations and Acronyms".
- "Payment Card Industry (PCI) Payment Application Data Security Standard" Version 3.2 October 2016".

These documents can be found at http://www.pcisecuritystandards.org

It is recommended to use the following reference material from the OpenWay documentation series:

- "WAY4™ Housekeeping".
- "Auditing Work with the Database in WAY4™".
- "WAY4 Audit Log Export".
- "Secure Access to Oracle Databases According to PCI DSS".
- "WAY4™ User Management".
- "Key Management in WAY4™".
- "Installing and Configuring the NetServer Java Secure Console".
- "Administering WAY4™ Application Server"

It is also recommended to refer to the following resource from Oracle's documentation series:

"Sustainable Compliance for the Payment Card Industry Data Security Standard".

This document is updated regularly (annually) and is also updated if changes in WAY4 are made that affect compliance with PA-DSS, or in the event of changes in PCI DSS or PA-DSS.

Document versions:

Document type	Version	Date of issue	Application version	PA-DSS Version	PCI DSS Version	Description of changes
Draft Version	0.1	28/04/2012	03.34.30	2.0	2.0	N/A
QA Review	0.2			2.0	2.0	N/A
Final Release	1.0	05/07/2012	03.34.30	2.0	2.0	N/A
Review for v. 03.35.30	1.1	07/09/2012	03.35.30	2.0	2.0	N/A
Review for v. 03.36.30	1.2	21/12/2012	03.36.30	2.0	2.0	N/A
Review for v. 03.37.30	1.3	14/01/2014	03.37.30	2.0	2.0	N/A
Review for v. 03.38.30	1.4	07/04/2014	03.38.30	2.0	2.0	N/A
Review for v. 03.39.30	1.4.1	01/09/2014	03.39.30	2.0	2.0	N/A
Review for v. 03.40.30	1.5	12/12/2014	03.40.20	2.0	2.0	N/A
Review for v. 03.41.30 and PA-DSS 2.1	1.6	03/12/2015	03.41.30	2.1	2.1	N/A
Review for v. 03.42.30 and some misprints corrected	1.7	04/02/2016	03.42.30	2.1	2.1	N/A
Review for PA-DSS 3.2	1.8	21/10/2016	03.42.3.x	3.2	3.2	Full review for PA-DSS 3.2 compliance
Correct misprints	1.9	04/04/2017	03.42.3.x	3.2	3.2	Path to nscipher.exe has been changed
Update	2.0	13/04/2017	03.42.3.x	3.2	3.2	Audit Logs
Update	2.1	19/04/2017	03.42.3.x	3.2	3.2	TLS 1.2 RC4 ban, mod_security is necessary PVV and Encrypted PIN are necessary
Review for PA-DSS 3.2	2.2	20/04/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x	3.2	3.2	Full review for PA-DSS 3.2 compliance
Review for PA-DSS 3.2	2.3	04/05/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x	3.2	3.2	Review for PA-DSS 3.2 compliance
Update	2.4	17/05/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x	3.2	3.2	Lifestyle Banking (%padss.application.mode=prod)
Review for PA-DSS 3.2	2.5	20/06/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x	3.2	3.2	Review for PA-DSS 3.2 compliance
Review for PA-DSS 3.2	2.6	03/07/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x	3.2	3.2	Disk Encryption, Health Monitoring over SSL

Document type	Version	Date of issue	Application version	PA-DSS Version	PCI DSS Version	Description of changes
Review for PA-DSS 3.2	2.7	17/08/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x 03.45.1.3.x	3.2	3.2	Added requirements: - prohibit automatic creation of Heap Dump. - encryption of the connection from the client browser to the IIS web server - hide wsruntime debugging information - mandatory setup of application authorization using client certificates. Appendix 1. Sample Key Custodian Form has been added.
Update	2.8	30/10/2017	03.42.3.x, 03.43.3.x, 03.44.1.3.x 03.45.1.3.x	3.2	3.2	- 3DES (112 bits) - AES (128 bits) - RSA (2048 bits)
Update	2.9	06/03/2018	03.44.1.3.x 03.44.2.3.x 03.45.1.3.x 03.46.1.3.x 03.47.1.3.x	3.2	3.2	Corrected key management requirements. Updated WAY4 Web logs file location.

Chapter 1. Sensitive authentication data

This chapter addresses compliance with PA-DSS Requirements 1.1.4 – 1.1.5 shown below.

PA-DSS Requirement	PA-DSS Topic
1.1.4	Delete sensitive authentication data stored by previous payment application versions.
1.1.5	Delete any sensitive authentication data (pre-authorization) gathered as a result of troubleshooting the payment application.

Definition of sensitive authentication data

PA-DSS considers sensitive authentication data to be the following:

1. Full contents of any track from a card's magnetic stripe (located on the back of the card, representation of any magnetic stripe, located on the card's chip).
2. Three-digit or four-digit card verification code, located on the front of the card or on the signature panel (CVV2, CVC2, CID, CAV2 data).
3. PIN and encrypted PIN block.

Important! The system cannot function normally without the PVV and Encrypted PIN fields. The fields contain PVV, IBM 3624 OFFSET, HSM PIN OFFSET (calculated with a Thales algorithm) and their storage by issuers or companies providing issuing services is not prohibited. PCI DSS does not prohibit storing these values in the database.

Characteristics of previous WAY4 versions

In previous versions of WAY4, sensitive data may have been stored in the APPL_CARD_INFO, CARD_INFO and MAILBOX database tables, since these versions did not support PCI DSS requirements and also resulting from migration of cards for which the card verification value had to be checked when the CVK value was unavailable. For compliance with PCI DSS, it is essential that data be deleted from the APPL_CARD_INFO, CARD_INFO and MAILBOX tables.

List of fields whose data must be deleted:

- APPL_CARD_INFO.CVC
- APPL_CARD_INFO.CVC2
- APPL_CARD_INFO.ICVV
- APPL_CARD_INFO.OFFSET_DATA
- APPL_CARD_INFO.PIN
- APPL_CARD_INFO.PVV

- CARD_INFO.CVC
- CARD_INFO.CVC2
- CARD_INFO.ICVV
- MAILBOX.BUF1
- MAILBOX.BUF10
- MAILBOX.BUF11
- MAILBOX.BUF12
- MAILBOX.BUF13
- MAILBOX.BUF14
- MAILBOX.BUF15
- MAILBOX.BUF16
- MAILBOX.BUF17
- MAILBOX.BUF18
- MAILBOX.BUF19
- MAILBOX.BUF2
- MAILBOX.BUF20
- MAILBOX.BUF21
- MAILBOX.BUF22
- MAILBOX.BUF23
- MAILBOX.BUF24
- MAILBOX.BUF3
- MAILBOX.BUF4
- MAILBOX.BUF5
- MAILBOX.BUF6
- MAILBOX.BUF7
- MAILBOX.BUF8
- MAILBOX.BUF9

Before deleting the CARD_INFO.CVC, CARD_INFO.CVC2, CARD_INFO.ICVV fields of the CARD_INFO table, additional analysis of the card verification methods used by the specific system instance is required. In the form "Full → Configuration Setup → Card Production Setup → Bank Production Parameters → Validation" ensure that "HSM both" and "HSM both YYMM" methods for checking cryptographic values are not used. If these methods are used, settings must be changed as follows:

- Instead of "HSM both", set "HSM both, no DB CVV2".

- Instead of "HSM both YYMM", set "HSM both YYMM, no DB CVV".

CVK keys must be available to verify values.

If the CARD_INFO.CVC, CARD_INFO.CVC2, and CARD_INFO.ICVV fields cannot be deleted because the values are used to verify CVV, CVC and similar values, if the key's appropriate validation property is missing (usually when a card has been moved from another system), cards must be reissued in WAY4. If they cannot be reissued, the guidance for PA-DSS Requirement 1.1 must be observed; namely, that it is permissible for issuers and companies that support issuing services to store sensitive authentication data if there is a business justification and the data is stored securely.

The CARD_INFO.PVV, CARD_INFO.OFFSET_DATA, and CARD_INFO.ENCRYPTED_PIN fields contain PVV, IBM 3624 OFFSET, and HSM PIN OFFSET and accordingly their storage by the issuer or company providing issuing services is not prohibited.

Data are deleted from database tables by running the appropriate scripts, provided on demand. Deletion of data can only be started after full completion of the current application workflow cycle and when online mode has been stopped. After deletion using DBMS tools, actions must be taken to prevent the deleted data from being recovered.

Lifestyle Banking

Previous versions of Lifestyle Banking may contain unmasked PAN in logs; therefore all log files from previous versions must be securely deleted. Log files are stored in folders (some may not exist in specific versions):

```
<Linux user home directory >/logs
```

To comply with PA-DSS requirements, an identifier (play.id) with the value "padss" has been created in the Lifestyle Banking configuration (play.id is a Play Framework attribute and makes it possible to create specific settings for different variants of Lifestyle Banking instances).

This identifier is used in the "application.conf" configuration file to set application parameters, including those used in the production system. For example, the IP address of the integration gateway (in the standard configuration, this is WAY4 Gate) compliant with PA-DSS requirements.

For operation in production mode, the following setting is also required:

```
%padss.application.mode=prod
```

Rainbow

Previous versions of Rainbow may contain unmasked PAN in message logs; therefore all log files from previous versions must be securely deleted.

Log files are stored in folders (some may not exist in specific versions):

```
<Customer Profile webapp>/WEB-INF/runtime/log
<Web Banking webapp>/WEB-INF/runtime/log
<Mobile Web Banking webapp>/WEB-INF/runtime/log
<Frontend Web Banking webapp>/WEB-INF/runtime/log
```

```
<Backend Web Banking webapp>/WEB-INF/runtime/log
```

Output of debugging information must be disabled (the value of the debug_enabled parameter in the ows-application.properties file must be false).

Processing sensitive data

If sensitive data must appear in logs, debugging files, in the database, or anywhere else for troubleshooting or debugging purposes, the following requirements for processing these data must be observed:

1. Sensitive data may be collected only when needed to solve specific problems that cannot be solved without collecting these data.
2. Sensitive data may only be stored in specific locations intended for this purpose, with restricted access.
3. Sensitive data may only be collected in the minimum amount needed to solve the problems described in Paragraph 1.
4. Sensitive data must be encrypted while stored, meaning a secure location must be used that is not linked to an operating system user record and password (see the document "Key Management in WAY4™").
5. Sensitive data must be securely and fully deleted immediately after use.

Chapter 2. Storing cardholder data and cryptographic keys

This chapter addresses compliance with PA-DSS Requirements 2.1. – 2.6 shown below.

PA-DSS Requirement	PA-DSS Topic
2.1	Securely delete cardholder data after customer-defined retention period.
2.2	Mask PAN when displayed so only personnel with a business need can see more than the first six/last four digits of the PAN.
2.3	Render PAN unreadable anywhere it is stored (including data on portable digital media, backup media, and in logs).
2.4	Protect keys used to secure cardholder data against disclosure and misuse.
2.5	Implement key management processes and procedures for cryptographic keys used for encryption of cardholder data.
2.5.1 - 2.5.7	Implement secure key management functions.
2.6	Provide a mechanism to render irretrievable cryptographic key material or cryptograms stored by the payment application.

Key-management requirements

Requirements for working with cryptographic keys used to encrypt cardholder data:

- Restrict access to keys and key components to the fewest number of custodians necessary. (Aligns with PA DSS Requirement 2.4).
- Store keys and key components securely in the fewest possible locations and forms. (Aligns with PA DSS Requirement 2.4).
- Implement key management processes and procedures for cryptographic keys used for encryption of cardholder data. (Aligns with PA DSS Requirement 2.5).
- Generate strong cryptographic keys (PA DSS Requirement 2.5.1). Please see current FIPS 140-2 recommendations for encryption algorithms and proper key lengths which are considered as strong.
- Secure distribution of cryptographic keys (PA DSS Requirement 2.5.2). Cryptographic key distribution is not necessary for normal processing.
- Secure storage of cryptographic keys (PA DSS Requirement 2.5.3). The recommended option for storage of a master key is inside the HSM. Working keys, encrypted by the master key, can be stored outside the HSM.

The storage of a master key inside Oracle Key Vault is also acceptable. Please refer to Oracle Database documentation for details.

- Implement cryptographic key change (PA DSS Requirement 2.5.4). Please see the text below for details regarding the requirement.
- Implement key retirement or replacement (PA DSS Requirement 2.5.5). Please see the text below for details regarding the requirement.
- Implement dual control cryptographic key management (PA DSS Requirement 2.5.6). The access password for the master key either for the PKCS#11 HSM token or for Oracle Key Vault must be password-protected using two components-passwords known by at least two different key custodians, respectively.
- Prevent unauthorized substitution of cryptographic keys (PA DSS Requirement 2.5.7). Each key management procedure must be performed using split knowledge of the master key storage password and under dual control and must be registered in the Key Management Log. Please refer to "Key Management in WAY4™" for more details.
- Use a mechanism to render irretrievable cryptographic key material or cryptograms stored by the payment application. (PA DSS Requirement 2.6) Standard TDE Resetting or Rotating (RE-KEY) operation is supported for keys stored in HSM storage (from Oracle database 11gR1) or in Oracle Key Vault. This process deactivates the previous TDE encryption key, creates a new TDE encryption key, and then activates it. The old key may be retained to allow decryption of previously encrypted data. This applies to TDE master and table (column) keys - they can be changed independently. Tablespace keys cannot be rekeyed, but as workaround data can be moved to a new encrypted tablespace. Please refer to Oracle Database documentation for details. Note that rotating column and tablespace keys may cause a significant performance overhead.

Key-management procedures must be performed pursuant to the document "Key Management in WAY4™".

Refer to the document "Key Management in WAY4™" for information on how to meet the requirements. It is mandatory to observe the main principles and procedures described in the aforementioned document for compliance with PCI DSS.

Any employee acting as a key custodian must fully understand and accept his/her key-custodian responsibilities, and this must be documented: each key custodian must fill in and sign a special form each time for each cryptographic key to which he/she has access. Requirements for the key management log can be found in the document "Key Management in WAY4™".

Each cryptographic key that is generated must be strong. The required key length according to industry standards and best practices, for example, is a minimum of 112 unpredictable bits for 3DES algorithm, 128 unpredictable bits for AES algorithm, and least 2048 unpredictable bits for the RSA asymmetric algorithm.

All cryptographic keys must be stored and distributed using a secure method only. For additional information, refer to the document "Key Management in WAY4™".

Cryptographic keys must be changed when they reach the end of their cryptoperiod. This period depends, in particular, on the following factors:

- Key length.
- Number of times the key was used.
- Type of encryption algorithm.
- Maximum defined period of time.

Detailed information about the cryptoperiod for a particular case can be obtained from industry standards and best practices, for example, NIST Special Publication 800-57.

An encryption key must be immediately retired and replaced when:

- The integrity of the key has been weakened.
- The key is compromised or suspected to have been compromised.
- The key has reached the end of its cryptoperiod.

For additional information about retiring and replacing keys, refer to the document "Key Management in WAY4™".

It is mandatory to observe key-management requirements for WAY4 compliance with PCI DSS.

Data retention

Cardholder data exceeding the customer-defined retention period must be securely deleted. Data whose retention period has expired must be deleted from the system. This requirement is mandatory for compliance with PCI DSS.

Pursuant to PCI DSS, data in WAY4 that must be deleted when their retention period expires are found in the following tables:

- ACNT_CONTRACT
- APPL_ACNT
- APPL_BATCH
- APPL_CARD_INFO
- APPL_CONTRACT
- APPL_INFO
- APPL_PAYM_REC_PARTY
- APPL_PM_KEYS
- BP_ATT
- BP_KEY

- BP_LOG
- BP_OP_LOG
- BP_PROCESS
- BP_QUEUE
- CARD_INFO
- CARD_STOP_LIST
- CARD_STOP_LIST_EXT
- COMS_LOG
- DOC
- DOC_MAILBOX
- INVOICE_PARTY
- LOG_FIELDS
- M_TRANSACTION
- MAILBOX
- NUM_REGISTERED
- ORIGINAL_DOC
- PAYM_REC
- PAYM_REC_PARTY
- PM_ADD_PARMS
- PM_KEYS
- PM_KEYS_OPT
- PM_TASK
- REMOTE_FILE_REQ
- SAFE_DOC
- TD_CONS
- TD_CONS_DATA
- TD_CONS_DOC
- TELEX_AUTH
- USAGE_HISTORY
- VOICE_AUTH
- XCHANGE_ACNT

Any historic data stored in the database, including the aforementioned, are automatically deleted from the system by Housekeeping procedures.

The document "WAY4™ Housekeeping" describes setting up and running procedures for automatically deleting data whose retention period has expired.

Housekeeping procedures must be run at least once a quarter.

Archived Housekeeping data must be stored in encrypted format in a secure location (see the document "Key Management in WAY4™".).

Housekeeping uses column encryption by default.

WAY4 processes clearing, dispute, and settlement files. These files contain cardholder data and must be deleted immediately after processing. Files must be securely deleted; i.e. after deletion, data must be rendered irretrievable even if they were stored on a RAM disk or in a secure location, since cardholder data must not be stored if they are not used. It is essential that clearing files, dispute files and settlement files be deleted for WAY4 compliance with PCI DSS. The exact location of files is determined by the relevant menu item configuration.

HCE Data Retention

In Host Card Emulation, data that pursuant to PCI DSS must be deleted when their retention period expires are automatically deleted by Housekeeping procedures. To define the retention period, use the "endOfLifeGraceDay" parameter from the "hce.json" configuration file.

Data storage requirements

Since cardholder information is stored in the database, critically sensitive data must be encrypted. Oracle Advanced Security Transparent Data Encryption (TDE) technology should be used to do so, and other requirements set forth in the section "Protect Cardholder Data" of the document "Oracle Database Security and the Payment Card Industry Data Security Standard" must be observed. It is imperative these requirements are observed for WAY4 compliance with PCI DSS.

Note that the use of TDE imposes additional hardware requirements for the database server.

When setting up TDE, define the following:

- Master key wallet location: on an external disk, on an internal disk or on a special security device.
- How to restrict wallet access privileges and prevent theft of the key.
- How to organize secure backup of the wallet; note that the wallet or its copy cannot be kept together with a database backup.

TDE technology can be used for:

- Column encryption.
- Encrypted tablespaces.

When the Oracle Partitioning option is absent (for example, in Oracle Database Standard Edition), encrypted disk partitions must be used to store data files, see the section "Encryption of disk partitions".

Column encryption

Before encrypting columns in the database, the settings specified in the section "Transparent Data Encryption" of the document "Oracle® Database Advanced Security Administrator's Guide" must be made.

To get the list of tables and columns whose data must be encrypted, run the script:

```
<OWS_Home>\db\ssp4 <OWS_Home>\install\tools\showEncryptedColumns.ssp
```

In the [Replacing] section of the DB.INI file created during initial installation of the system, specify the parameters:

```
encrypt=encrypt no salt 'nomac'
encryptLob=encrypt
```

If DB.INI parameters were specified before the system was installed, data will already be encrypted. Otherwise (and also after system upgrade if additions were made to the list of encrypted columns) the Online Table Migration Tool, described in \manuals\system\ows_OnlineTableMigration.html must be used for encryption.

In addition, the following must be executed in the Oracle database for LOB encryption:

```
alter system set DB_SECUREFILE = ALWAYS
```

Encrypted tablespaces

To use encrypted tablespaces, the encrypt and encryptLob values must be removed, and tablespace mapping with the _E suffix (encrypted, for example, OWLARGE_E_D) specified instead so they refer to encrypted Oracle tablespaces created earlier by the administrator. To create mapping, copy existing mapping for all tablespaces except OWTEMP, adding the _E suffix to each line:

```
# early present lines
OWLARGE_D=LARGE_D
OWLARGE_I=LARGE_I
# mapping added for Tablespace Encryption
OWLARGE_D=LARGE_D
OWLARGE_I=LARGE_I
OWLARGE_E_D=LARGE_ENC_D
OWLARGE_E_I=LARGE_ENC_I
```

If DB.INI parameters were specified before installing WAY4, data will already be encrypted. Otherwise (and also after system upgrade if additions were made to the list of encrypted columns) the Online Table Migration Tool, described in \manuals\system\ows_OnlineTableMigration.html must be used for encryption.

Configuring scripts when using TDE

The following scheme is recommended when TDE (Transparent Data Encryption) is used:

- For encryption, it is recommended to use a separate "Encryption Wallet" without the "auto-login" function. This increases security; since to open the "Wallet", an additional password must be entered each time the database is started.
- To store (hide) user names and passwords, another "Wallet" can be used with the "auto-login" function (since scripts are only executed by an Oracle user).

"Wallet" must be created as follows:

- Make all settings necessary for encryption. This includes creating a "Wallet" for encryption (Encryption "Wallet" is created automatically during execution of the SQL expression "alter system set encryption key identified by <wallet_password>"). It is recommended to check encryption parameters on a test system (preferably with restart of the database).
- Create a "Wallet" for storing script passwords. "Auto-login Wallet" is created automatically using the mkstore program. "Auto-login Wallet" is created after all encryption settings have been made (and checked).

This procedure makes it possible to avoid situations when Oracle adds encryption of an "Auto-login Wallet" intended for storing script passwords.

Make sure that privileges for directories and for files created in these directories during the "Wallet" creation process have been granted to the operating system's appropriate users (Oracle in Linux, SYSTEM in Windows). In a number of cases, privileges for cwallet.sso and ewallet.p12 files for "Wallet" with "Auto-login Wallet" must be created manually for SYSTEM.

Error 9215461 (ORA-28362) occurs when "Encryption Wallet" is deleted and re-created. This error should have been fixed by Oracle in version 11.2.0.3. For this reason, it is necessary to create settings correctly the first time and it is recommended to correctly create "Wallet", the key and password for encryption.

Housekeeping and TDE

If column encryption is used for WAY4 tables, no additional steps are required to set up Housekeeping (HSK): archive tables created by HSK will have the same encrypted columns as the original WAY4 tables.

If tablespace encryption is used in WAY4, HSK must be set up for encryption of archive tablespaces:

1. If tablespaces are created automatically, HSK parameters must be set in Housekeeping\Configuration \Tablespace Group -> Tablespace Parameters:

```
DATAFILE '...' SIZE ... ENCRYPTION USING 'ENCRYPTION_ALGORITHM' DEFAULT
STORAGE(ENCRYPT)
```

where ENCRYPTION_ALGORITHM is AES128. Other algorithms are also supported (AES256, AES192, 3DES168).

6. If HSK uses manually created tablespaces, the database administrator must create these tablespaces manually.

```
ENCRYPTION USING 'ENCRYPTION_ALGORITHM'
```

```
DEFAULT STORAGE(ENCRYPT)
```

Specify parameters in Housekeeping \Configuration\Tablespace Group -> Tablespace Parameters -> Tablespace Mask.

Encryption of disk partitions

Disk partitions are encrypted by special utilities. The example below shows use of the cryptsetup utility in Linux to encrypt disk partitions for storing WAY4 tablespace data files.

Creation of an encrypted partition:

```
cryptsetup -v luksFormat /dev/sdc1
```

Opening the encrypted partition and assigning an alias:

```
cryptsetup luksOpen /dev/sdc1 sdc1e
```

- Oracle ASM configuration:
  - Group Name – DATAE.
  - Disk – /dev/mapper/sdc1e.
  - Redundancy = External.
- WAY4 tablespace data files are created in the Oracle ASM – DATAE group.

Data storage requirements

Exchange files must be archived regularly and stored in encrypted format.

Reports generated in the system must be archived regularly and stored in encrypted format.

Data obtained when troubleshooting must be securely deleted immediately after the necessary procedures have been performed.

Swapping in systems running under Java components handling CHD must be disabled or an encrypted disk must be used for swapping.

HCE data storage requirements

Host Card Emulation doesn't have a DB Server, but contains persistent data on the encrypted disk of each node (see the section "Preparing Protected Disk Space" of the document "WAY4™ Host Card Emulation Installation and Setup").

Database server requirements

If queries to the Oracle DBMS must be traced, this must be done without saving bind variable values. Traces with bind variable values can only be used on a test system. It is essential this requirement be observed for WAY4 compliance with PCI DSS.

Requirements are observed pursuant to the document "Oracle Database Security and the Payment Card Industry Data Security Standard".

User workstation requirements

Tracing must be disabled on workstations.

WAY4 product and component compliance with PCI DSS

The table below contains information about compliance with requirements for storing and logging critically sensitive information by WAY4 products and components.

Name of product or component	**Compliance**
Datamart	PAN is masked after the first 6 and to the last 4 digits in the Datamart database; therefore it is impossible to get unmasked data from Datamart.
WAY4U SMS Banking	Data are masked in logs. Data masking is enabled by default by the parameter log_filtering_enabled=yes in the "WEB-INF/config/work/ows-application.properties" file. **Important!** Disabling this parameter will result in noncompliance with PCI DSS.
WAY4 manager	PAN is always masked in all "read only" forms and in edit forms used in "View" mode. If a form allows PAN to be edited, PAN becomes unmasked when focus is set on the field. The menu available for a certain employee must be configured according to this employee's business needs. When writing to a log, all number sequences longer than 6 digits are masked.
Remote access	When writing to a log, all number sequences longer than 6 digits are masked.
e-Commerce issuing, e-Commerce acquiring, Bill payments	Data are always masked in logs. It is not possible to save unmasked data in logs.
Clearing Files	A RAM disk or encryption in a secure location is required for file storage (see the section "Secure Storage Key Management" of the document "Key Management in WAY4™"). Files imported to WAY4 or sent to a recipient must be immediately deleted from the disk. **Important!** It is essential these requirements be met for WAY4 compliance with PCI DSS.
Application Server	Data are not stored and are not logged.
Payment Server	Data are not stored and are not logged.
Reporting	Data are masked in logs. By default, report tracing is disabled. **Important!** Enabling report tracing will result in noncompliance with PCI DSS.

Name of product or component	**Compliance**
File Exchange Engine (pipes)	Data are masked in logs. Disk encryption is required for storage. Java pipes save information in a standard WAY4 Manager log; therefore it is impossible to get logs with unmasked data for Java pipes. C pipes: to disable data masking, the "NOMASK_TRACE_START" pipe parameter can be used. **Important!** Enabling this mode results in noncompliance with PCI DSS.
WAY4 Web	Masking of stored data is enabled by default. For debugging purposes, card number masking in logs can be temporarily suspended for 15 minutes in a running instance of the server by the console command "<WS_Runtime_Path>\WEB-INF\commands\ Logging\Filter\suspendMaskingMode". **Important!** Enabling this mode results in noncompliance with PCI DSS.
Transaction Switch	Data are masked in logs.
NetServer	Data are masked in logs.
Access server	Data are masked in logs.
NetServer Console	Data are masked in logs.
Rainbow	Data are masked in logs.
Lifestyle Banking	Data are masked in logs.
Host Card Emulation (HCE)	Data are masked in logs.
Authentication Server and Data Gate	Incoming connections to AuthServer use TLS 1.2, AuthServer connections to Data Gate use the Secure ISO interface. These options are enabled by default after installation. Changing the parameter will result in noncompliance with PCI DSS.

Chapter 3. Registering users and setting passwords

This chapter addresses PA-DSS Requirements 3.1 - 3.2 shown below.

PA-DSS Requirement	PA-DSS Topic
3.1	Use unique user IDs and secure authentication for administrative access and access to cardholder data.
3.2	Use unique user IDs and secure authentication for access to PCs, servers, and databases with payment applications.

Default administrative accounts

Default administrative accounts created during installation cannot be used to access the payment application, for example, administrative accounts (such as "sys") cannot be used for access to data.

All default administrative accounts must be assigned secure authentication policies (even if the accounts are not used), and then all unused administrative accounts must be deactivated.

Whenever possible, user accounts must be assigned secure authentication policies. This concerns both the payment application and the environment in which it operates (OS, servers, etc.).

NetServer

Setup of administrative and user accounts is a mandatory step of the payment application installation procedure. This procedure is described in the document "Installing and Configuring the NetServer Java Secure Console".

Transaction Switch

The WAY4 Transaction Switch application is installed on the WAY4 Application Server platform and is managed by a web console. Information about setting up console user access is described in the section "Managing WAY4 Applications" of the document "Administering WAY4™ Application Server".

Principles of secure authentication

This section addresses compliance with PA-DSS Requirements 3.1.1. – 3.1.11.

To comply with these requirements, user authentication for access to the payment application, including access to all workstations, servers, and databases must meet the following requirements:

1. The use of shared identifiers for access (accounts and passwords) is prohibited, including for administrative purposes.

Each user must be given a unique value as a first-time password. Default system functionality must also be used that requires a password to be changed the first time it is used.

2. User passwords must meet the following requirements:
   - User passwords must be changed at least every 90 days (password life time).
   - Be at least 7 characters in length.
   - Contain both numeric and alphabetic characters.
   - A new password must de different than any of the last four passwords used (password reuse max).
   - A limit is set (no more than 6 times) for the number of attempts to enter the wrong password after which the account is locked out for at least 30 minutes or until the administrator enables the account (password lock time).
3. Client applications are automatically locked out after no more than 15 minutes of idle time.

Functionality for locking out unused accounts must be used (see the section "Locking Inactive Accounts" of the document "WAY4™ User Management"). Unused accounts must be locked out for compliance with PCI DSS.

The Oracle DBMS allows password requirements to be verified using standard SQL script. This script must be used to ensure passwords comply with PA-DSS. Rules for using the functions are described in Oracle "Oracle® Database Security Guide" documentation in the sections "Authentication Methods" and "Password Complexity Verification".

Password quality must be checked automatically after generation using the Oracle function "PASSWORD_VERIFY_FUNCTION" function. "PASSWORD_VERIFY_FUNCTION" lets a PL/SQL password complexity verification script be passed as an argument to the CREATE PROFILE statement.

Example of changing the default profile:

```
ALTER PROFILE "DEFAULT" LIMIT
 PASSWORD_LIFE_TIME  90   <-- Renewal of password every 90 days
 PASSWORD_GRACE_TIME 10   <-- Give the user 10 days' grace period
 PASSWORD_REUSE_MAX  5    <-- Ensure at least 5 different passwords before
reusing a password
 FAILED_LOGIN_ATTEMPTS 6  <-- Allow 6 attempts to login
 PASSWORD_LOCK_TIME .0208 <-- After 5 unsuccessful login attempts, lock
account for 30 min, then allow new attempts
 PASSWORD_VERIFY_FUNCTION ora12c_verify_function; <-- enforce password
complexity as desired
```

Where "PASSWORD_VERIFY_FUNCTION" is "ora12c_verify_function" (<ORACLE_HOME> /rdbms/admin/utlpwdmg.sql). The function is provided by Oracle. There are also two functions "verify_function" (10g)

and"verify_function_11G" (11g). For 12c there are four new functions: "ora12c_verify_function" and "ora12c_strong_verify_function" and the auxiliary functions "complexity_check" and "string_distance".

Passwords for database administrator accounts must meet quality (complexity) requirements.

Since the WAY4 database schema owner's account is only used during system installation and upgrade, this account must be locked out at all other times.

Changes in security settings applicable to unique user accounts and weakening of authentication policies compared to those recommended will result in noncompliance with PCI DSS.

NetServer

Java Secure Console supports two different user roles: "Active" and "Passive". The "Active" role allows a user to manage the payment application. The "Passive" only allows the application's status to be monitored. After a timeout (by default, 10 minutes) console mode will be switched from an "Active" user role to the "Passive" role.

Passwords for administrator accounts must meet the following requirements:

- Minimum length of at least 7 characters.
- Contains both numeric and alphabetic characters.
- Must not be the same as any of the last 4 passwords used.
- No more than 6 attempts to enter the password.

Messenger banking

The length of a user session is defined by the session_expiry_sec parameter (ows-application.properties file) and must not exceed 300 seconds.

Transaction Switch

For information about setting up user access to the console for managing Transaction Switch, see the section "Managing WAY4 Applications" of the document "Administering WAY4™ Application Server".

Rainbow Banking

The following files containing WAY4 DB access parameters must be encrypted using the nscipher utility (see the section "Encrypting configuration parameters"):

```
<Customer Profile webapp>/WEB-INF/config/work/ows-application.properties
<Web Banking webapp>/WEB-INF/config/work/w4c.properties
<Mobile Web Banking webapp>/WEB-INF/config/work/w4c.properties
<Telegram Bot webapp>/WEB-INF/config/work/ows-application.properties
```

Encrypting configuration parameters

Some parameters in configuration files must be encrypted. This is specified in comments to the parameter. For example:

```
password="encrypted:specify password encrypted by nscipher.exe"
```

The "password" parameter with an encrypted value may appear as follows:

```
password="encrypted:5CD2466B4D8D25ED0B05DBBFFFFC8A81B4D5A640A7D356"
```

The value after "encrypted:" is the encrypted value.

Use the ns_cipher utility to get an encrypted value. The utility is included in the WAY4 Application Server distribution:

```
<APP_SERVER>\appserver\bin\tools\nscipher.exe
```

The utility is run from the command line. Command format:

```
nscipher.exe ows_application > <path to the file (full or relative) to
which the encrypted password will be written >
```

For example:

```
nscipher.exe ows_application > c:\pass.txt

nscipher.exe ows_application > .\out\pass.txt
```

After it has been started, the utility asks for a password, creates its encrypted value and saves it as a string in the "pass.txt" file (value specified in the command line). The path specified in the command line must exist. If it doesn't, the utility will return the error: "The system cannot find the path specified".

Data access requirements

It is strongly recommended to use the user password encryption function built into client applications supplied with WAY4 (see Limiting data access with user password encryption" in the document "WAY4™ User Management").

The use of third-party applications for access to database data is strictly prohibited.

Oracle Database provides network encryption and strong authentication.

Database server requirements

Unused DBMS user accounts, including administrative and service accounts must be locked out or deleted. Access passwords, including default passwords and those created during software installation, must be changed for administrative accounts that are in use.

The recommendations set forth in the document "Sustainable Compliance for the Payment Card Industry Data Security Standard" must also be observed.

User workstation requirements

Client applications must be automatically locked out after no more than 15 minutes of idle time.

Chapter 4. Audit log

This chapter addresses PA-DSS Requirements 4.1 – 4.4. shown below.

PA-DSS Requirement	PA-DSS Topic
4.1	Implement automated audit trails.
4.4	Facilitate centralized logging.

Starting from WAY4 version 03.34.30, a PA-DSS-compliant audit log is automatically maintained. The log is maintained in the SY_AUDIT_LOG table. Audit logs are a mandatory condition for a WAY4 instance to comply with PCI DSS. By default, audit logging is enabled. The menu item "Full → DB Administrator Utilities → Users & Grants → Audit Log" is used to view the audit log.

The following information is stored in this log:

- User identifier, additionally (if applicable) with the IP address indicated.
- Type of event.
- Date and time of event.
- Success or failure indication.
- Confirmation that the event occurred.
- Identifier or name of affected data, system object, or resource.

The audit log can be used to reconstruct the following events:

1. User actions to access card data, i.e.: open forms with card data, call processes that have access to card data.
2. All actions taken by administrators as provided for in the application as well as those saved in the Oracle audit log (for more information, see "Database server requirements").
3. Access to the audit log from the application.
4. Invalid attempts to access card data.
5. All attempts (successful and failures) to login to the application.
6. All attempts to initialize the audit log through the application.
7. Creation and deletion of system objects (users, forms, processes, menus, etc.) through the application.
8. User actions for starting/stopping the application and its components.
9. Request for access to the application audit log.
10. Administrative actions with the audit log.

Audit logs must always be enabled and disabling them will result in noncompliance with PCI DSS.

NetServer

The NetServer audit log is maintained in compliance with PA-DSS. The log is kept in the "action.log" file. An audit log must be kept for a NetServer instance to comply with PCI DSS.

Audit logs must always be enabled; there should be no configuration options allowing audit logs to be disabled.

Audit log files cannot be deleted or changed using Java Secure Console.

Transaction Switch

For information, refer to the section "Viewing and Analysing Log Files" in the chapter "Managing WAY4 Applications" of the document "Administering WAY4™ Application Server".

Support of centralized logging

To support third-party centralized logging systems and for automatic backup of the audit log, there is a special process in WAY that copies audit log data to a file. This file is in text format; fields in the file are separated by tabs, allowing audit log data to be exported to practically any third-part centralized logging system.

To limit access to these files, it is recommended to copy audit log data to media, for example, to a file server, with limited employee access. In particular, this list must not include database administrators and WAY4 users with administrative privileges.

To start the process of copying audit log data, the menu item "Full → DB Administrator Utilities → Users & Grants → Dump Log is used (for more information, see the document "WAY4 Audit Log Export"). Copying must be performed regularly; its frequency is determined by the client's security policies. To automate this process, it is recommended to use Scheduler.

NetServer and Transaction Switch, Lifestyle Banking and Rainbow Banking require creation of a special process for centralized logging and backup of the audit log.

There are two options for centralized logging:

- Copying audit log files from a local computer to a remote computer.
- Use of a remote file system.

In the first option, the system operates when Log Server is not available. In the second case, the ability to write to audit log files when Log Server is unavailable must be clarified.

Audit log files created by all WAY4 applications are in text format (file fields are separated by tabs) allowing audit log data to be exported to practically any centralized logging system.

Copying audit log files from a local computer to a remote computer

1. There must be a computer with RHEL for storing audit log files in Internal Network, hereinafter Log Server.
2. SSH key access has been set up (this mechanism must be used because secure data transmission is required).
3. Log Server establishes an SSH connection with Frontend Server (CRON job, executed once an hour) and executes RSYNC for the directory with audit log files (see "http://troy.jdmz.net/rsync/index.html").
4. Example of the script:

```
# Linux get logs script
#---------------------------------
RHOST=server1
RUSER=way4
RPATH=/home/way4/appserver
APPLS="frontend content way4u"
LPATH=/opt/all_logs
#---------------------------------
RSYNC=/usr/bin/rsync
SSH=/usr/bin/ssh
#KEY=/home/way4/.ssh/id_rsa
OPTS="-azru -e"
#---------------------------------
if [ ! -d $LPATH ]; then
  mkdir -p $LPATH
fi
date
for APPL in $APPLS ; do
  if [ ! -d $LPATH/$APPL ]; then
    mkdir -p $LPATH/$APPL
  fi
  $RSYNC $OPTS $SSH $RUSER@$RHOST:$RPATH/applications/$APPL/logs
$LPATH/$APPL


  if [ ! -d $LPATH/$APPL ]; then
    mkdir -p $LPATH/$APPL/runtime
  fi
  $RSYNC $OPTS $SSH
$RUSER@$RHOST:$RPATH/applications/$APPL/webapps/$APPL/runtime/logs
$LPATH/$APPL/runtime
done
#$RSYNC $OPTS "$SSH -i $KEY" $RUSER@$RHOST:/logs $LPATH
```

5. Audit log files on Log Server are analyzed and deleted (CRON job). Audit log files on Frontend Server are deleted automatically by the application. Example of deleting all files over 30 days old in the "/opt/all_logs" directory:

```
# Deletes files in /opt/all_logs older than 30 days
find /opt/all_logs/ -mtime +30|xargs rm -f
```

6. Protection of log files from modification by any application other than WAY4U ("appserver/jdk/current/bin/java") under a specific user for RHEL will be implemented using SELinux (http://www.linuxtopia.org/online_books/rhel6/rhel_6_selinux/, http://www.linuxtopia.org/online_books/rhel6/rhel_6_confined_services/rhel_6_services_sect-Managing_Confined_Services-rsync-Booleans.html). At the present time, this guide is not ready.
7. Log files are not rewritten if their size exceeds a specific value, since according to the previous paragraph it is difficult or impossible to modify files.

Use of a remote file system

1. The requirements of http://www.cyberciti.biz/tips/rhel-centos-mounting-remote-filesystem-using-sshfs.html must have been met. This mechanism is used since secure data transmission is necessary.
2. Frontend Server directories on the remote server have been mounted.
3. Files have been protected and deleted pursuant to paragraphs 3 and 4 of the section "Copying audit log files from a local computer to a remote computer".

Database server requirements

Audit is performed by operating system tools. An audit log is kept for at least three months. Old logs are exported and stored together with other historical data.

Oracle DBMS tools are used for required auditing, as described in the document "Auditing Work with the Database in WAY4™". It is recommended use operating system tools to audit access to Oracle audit log files.

Note that disabling or failure to audit with Oracle and operating system tools results in noncompliance with PCI DSS.

File server requirements

Audit is performed by operating system tools. Audit logs are kept for at least three months. Old logs are exported and stored together with other historical data.

Application logging

WAY4 components maintain their own logs. For purposes of centralization, the log format for all components is simple human-readable text format; each log record is distinguished from the previous one by a CR/LF sequence.

The following table shows the location of logs:

Name of product or component	**Log file location**
Datamart	A separate log is not used.
WAY4U SMS Banking	Log files are stored in the directory <webapp>\WEB-INF\runtime\log\.
WAY4 manager	Log files are stored in the directory USER_HOME/.OWS/PROFILE_NAME/log.
Remote access	Log files are stored in the directory USER_HOME/.OWS/PROFILE_NAME/log.
e-Commerce issuing, e-Commerce acquiring, Bill payments	Log files are stored in the application's "log" directory.
Clearing Files	A separate log file is not used; the File Exchange Engine log is used.
Application Server	A separate log is not used.
Payment Server	A separate log is not used.
Reporting	A separate log is not used; Oracle Reports tracing is used.
File Exchange Engine (pipes)	Java pipes store information in the standard WAY4 Manager log. C pipes: the log file path is set by the "TRACE" pipe parameter.
WAY4 Web	WS Runtime server log files are stored in the "<WS_Runtime_Path> \WEB-INF\logs\" directory. WS Runtime server settings are stored in the "<WS_Runtime_Path> \WEB-INF\conf\" directory. IIS (Internet Information Services) log files are located in the "<local disk>/inetpub/logs/logFiles" directory. Log files for errors that occurred during operation of the WAY4 Web Site are located in the "<path to site>/App_Data/ErrLog" directory, where <path to site> is the "installation_name" directory located in the "install_dir" directory (parameters are described in the section "[Common] Section" of WAY4 Web documentation). When there are no errors, the "ErrLog" directory will not be created. Error log files can also be analyzed through the web browser, specifying the URL address "http://<site_host>:<site_port>/elmah.axd".
Transaction Switch	Log files are stored in <webapp>\WEB-INF\logs\
NetServer	Log files are stored in the directory where the application is installed, with access limited by operating system tools.
Access server	Log files are stored in the directory where the application is installed, with access limited by operating system tools.
Java Secure Console	Log files are stored in the directory where the Java Secure Console application is installed, in the "logs" subdirectory.
Lifestyle Banking	Log files are stored in <Linux user home directory>/logs

Name of product or component	**Log file location**
Rainbow Banking	Log files are stored in: <Customer Profile webapp>/WEB-INF/runtime/log <Web Banking webapp>/WEB-INF/runtime/log <Mobile Web Banking webapp>/WEB-INF/runtime/log <Telegram Bot webapp>/WEB-INF/runtime/log
Host Card Emulation	The log is stored in: "%GRID_HOME%/owwork/logs/{date,yyyy-MM-dd~HH.mm}-gigaspaces-{service}-{host}-{pid}.log"

Note. The %GRID_HOME% environment variable will be set after GS Bootstrap is started the first time.

Chapter 5. Developing secure payment applications

This chapter addresses PA-DSS Requirements 5.1, 5.2.5, 5.2.9, 5.2.10 and 5.4.4 shown below.

PA-DSS Requirement	PA-DSS Topic
5.1	Develop secure payment applications.
5.2.5	Prevention of information leakage about applications configuration, their internal workings through improper error-handling methods.
5.2.9	Cross-site request forgery (CSRF).
5.2.10	Session management.
5.4.4	Implement and communicate application versioning methodology.

Application requirements

This section addresses compliance with PA-DSS Requirement 5.1.

WAY4 Web

Developer info must not be provided to users (web.config file):

```
<add key="EnableDeveloperInfo" value="false" />
```

In production systems, WS Runtime debugging information in system logs must be hidden. Ensure the sql_debug configuration parameter value (Enable/Disable including SQL-query of service into response) is "no".

```
<options
    sql_debug="no"
...
```

The sql_debug parameter is specified in the file:

```
%WAY4ApplicationServer%\appserver\applications\wsruntime_xxx\webapps\wsrunt
ime_xxx\WEB-INF\conf\global-options.xml
```

Messenger banking

This section addresses compliance with PA-DSS Requirement 5.2.5.

There must be no open IP addresses in the Messenger banking configuration.

Rainbow Banking

This section addresses compliance with PA-DSS Requirements. 5.2.9 – 5.2.10.

For protection from Cross-site request forgery (CSRF) attacks, the value of the csrf_protection parameter in the ows-application.properties file must be true.

The ability of a user to work simultaneously in several http sessions must disabled (in the ows-application.properties file, the allow_multiple_user_sessions parameter value must be false).

WAY4 versioning methodology

This section addresses compliance with PA-DSS Requirement 5.4.4.

A WAY4 version number must have the following format:

GG.MM.p.m.bbbb

Where:

- GG – two digits indicate the system generation number, they change (increase by 1) if global system architecture changes.
- MM – two digits indicate the number of the major functional release, the number increases by 1 if major functional changes are implemented. A major functional release can contain impact changes according to PA-DSS classification.
- p – one digit indicates a PA-DSS release, the number increases by 1 if impact changes according to PA-DSS classification are implemented. Reset after each major functional release.
- m – one digit indicates the number of the minor functional release, the number increases by 1 if minor functional changes are implemented. Reset after each major functional release. Can contain No Impact changes according to PA-DSS classification.
- bbbb – four digits indicate the build number, the number increases by 1 when each system build is released with bugfixes and patches. Reset after each minor functional release. Can contain No Impact changes according to PA-DSS classification.

Important! This format is used starting from version 03.44. Previously, up to version 03.43, the following format was used:

GG.MM.mb.bb

Where:

- G – Generation Number
- M – Major Functionality Number
- m – Minor Functionality Number
- b – Build Number

Wildcards

A wildcard can be used to indicate a group of major releases and to group minor releases and build numbers.

Only the bbbb component of the version string can be replaced with a wildcard symbol to indicate a wildcard version (GG.MM.p.m.**x**). For example: 03.44.1.3.0006. Such wildcard version is used to indicate a group of minor functional releases without any changes that may affect security or PA-DSS Requirements implementation. Only changes with no impact to security or PA-DSS requirements implementation are allowed within same wildcard version.

Chapter 6. Use of wireless data transmission technologies

The WAY4 application is not designed for use with wireless networks and all network communications between WAY4 components have to be done over wired networks only. Use of wireless networks for communication of WAY4 components is prohibited and all wireless network interfaces on all system and network components handling internal WAY4 communications should be disabled.

If there are any wireless networks within the organization they should be either air-gapped from networks handling internal WAY4 communications or a firewall be in place between any wireless networks and networks handling internal WAY4 communications and permit only explicitly authorized traffic between the wireless environment and networks handling internal WAY4 communications. For all wireless networks within the organization (even if they are not handling internal WAY4 communications) the following should be in place

- All default wireless encryption key passwords and SNMP strings should be changed on installation.
- Wireless encryption keys, passwords and SNMP strings should be changed anytime anyone with knowledge of the keys/passwords leaves the company or changes positions.
- Industry best practices, (for example, IEEE 802.11.i - Wi-Fi Protected Access II / WPA2) should be in use to provide strong encryption for authentication and transmission.

Chapter 7. Security patches

This chapter addresses PA-DSS Requirement 7.2.3 shown below.

PA-DSS Requirement	PA-DSS Topic
7.2.3	Provide instructions for customers about secure installation of patches and updates.

Hotfixes are used to distribute WAY4 security patches.

Requirements

1. PGP keys must be installed to encrypt and sign data being transmitted between the client and Customer Support.
2. Personnel that can be involved in the process of receiving and installing a Hotfix on behalf of the Customer must be registered in Customer Support's CRM as official users. The Customer must support up-to-date information about its users specified in the registration form, and is obligated to duly inform Customer Support of changes in user information.
3. WAY4 as well as all its updates and patches are distributed using individual FTP connections on the client side, secured by PGP.
4. An FTP connection is provided on request and deactivated immediately after use.
5. WAY4 components are not updated and are not patched by providing access to them.
6. In WAY4, the remote access client application and patches for it are distributed securely. Security is provided using an electronic signature that is automatically checked during installation.
7. In NetServer, the remote access client application and patches for it are distributed securely. Security is provided using an electronic signature that is automatically checked during installation.

Procedure

1. Customer Support prepares a Hotfix, encrypts and signs the Hotfix using the specific PGP key for each Customer and puts the Hotfix on FTP into dedicated directories for each Customer.
2. Customer Support informs registered users about the Hotfix by e-mail or phone.
3. The Customer's registered user uploads the Hotfix from FTP, decrypts it and verifies the signature. If problems occur with decryption or verification of the signature, the registered user must immediately notify Customer Service of the incident and must stop processing the Hotfix (the Hotfix

must not be installed). The incident must be investigated by the Customer and Customer Support.

4. If the Hotfix has the correct signature and can be successfully decrypted, it must be installed according to the instructions for the Hotfix.

Installation process overview

Note that in various solutions, WAY4 is provided as a specific set of components. This set of components may differ from solution to solution, depending on required functionality.

For more information about the installation process, it is recommended to use the following resources from the OpenWay documentation series:

- "Administering WAY4™" Application Server" (WAY4_Application_Server_Administering.pdf).
- "OpenWay Upgrade Manual" (Doc\upgrade.html).

WAY4 is a multicomponent system. Installation of WAY4 consists of the following steps:

- Planning installation. The installation procedure must be documented with consideration of the system instance's specific nature. Installation log files must not be lost during installation. The installation plan must be aligned with WAY4 documentation (installation manuals, requirements, release notes).
- Completion of pre-installation tasks. Before starting installation of WAY4, the Customer's working environment and organizational support must meet the requirements of OpenWay personnel. All networks must be fully configured by this time.
- Installation and setup of the Oracle DB (database, Oracle Partitioning, Advanced Security Option, software for backup copying).
- Workstation setup.
- Installation and setup of Oracle-based components (Cards, Datamart).
- Troubleshooting after each component has been installed.
- Installation of components based on the Cards platform and Datamart platform.
- Troubleshooting after each component has been installed.
- Installation of Application Server.
- Troubleshooting after each component has been installed.
- Installation of components based on the Application Server platform.
- Troubleshooting after each component has been installed.

Chapter 8. Payment application security

This chapter addresses PA-DSS Requirement 8.2 shown below.

PA-DSS Requirement	PA-DSS Topic
8.2	Use only necessary and secure services, protocols, components, and dependent software and hardware, including those provided by third parties.

List of third-party software used, system services and protocols used

Name of product or component	Third-party software	System services	System protocols
WAY4 Cards	For the DB server: Oracle DB Server M v. 11g For the file server: RAM Disk or Encrypted Container at the Customer's discretion.	Not used	DB server TCP/IP, port for Oracle Listener defined in the Oracle configuration CIFS (Windows), SAMBA (Linux) for the file server
WS Runtime	Not required	Not used	HTTP or HTTPS (1 port for accepting WS Runtime external requests), SQL-NET (1 for connection with the DB)
Payment Server	Java Virtual Machine v. 7.X or newer, WAY4 Application Server	Not used	TCP/IP, the port must be set in the configuration
Billing Gateway	Not required	Not used	TCP/IP, the port depends on Billing Provider and must be defined in the configuration
WAY4 Web	Client java packages (jars)	Not used	HTTP, HTTPS (port 443)
WAY4 Manager Client	JRE 1.6 (in HOME) and Oracle client (to change passwords) JRE 1.6 (in HOME), Oracle client (the version must be the same as the DB version)	Not used	TCP/IP (connection with the Oracle DB, host and port are defined by the administrator in the *db.ini* file)
Reports	Oracle Application Server v. 11 with Oracle Reports Services v. 11g	Not used	HTTP/HTTPS

Name of product or component	Third-party software	System services	System protocols
Remote Access Server	Oracle DB Client (the version must be the same as the DB version)	Not used	TCP/IP/ HTTP/ Oracle OCI Ports: 1) Inbound TCP port, selected when the application was created (for example, 8080) 2) Oubound DB port, selected in Oracle Connection Manager or DB Listener port (for example, 1521)
Datamart	Oracle DB Server for the DB server v. 11g	Not used	TCP/IP, port for Oracle Listener, set in the Oracle configuration
File Exchange Engine (Pipes)	JAVA Virtual Machine latest version of v. 7.X	Not used	TCP/IP, port set in the DB.ini file, must correspond to Oracle Server settings
3-D Secure	Java, Apache Java – latest version of v. 7.x, Apache – latest version	Not used	TCP/IP, SSL Ports: IP HTTPS(443) + MasterCard/VISA DS (defined during implementation)
POS Management Server	Not required	Not used	TCP/IP, the port depends on the POS network and must be configured
Application Server	Apache Web Server 2.2.22, Apache Tomcat 6.0.35, JDK 1.7 (on AIX PPC, Oracle SPARC, Linux i386/x64, Windows i386), ANT 1.8.2 ModSecurity www.modsecurity.org	Not used	TCP/IP Web Server ports: Inbound HTTP (8080 by default) Inbound HTTPS (8443 by default) The customer can change these ports in the Web Server configuration
WAY4U SMS Banking	Java Virtual Machine latest version of v. 7.x	Virtual COM port, if GSM modem is used	TCP/IP, the port is defined in the configuration, HTTP (if an HTTP channel is used), SMPP (if an SMPP channel is used)
NetServer	Java SDK	Network Service	TCP/IP
Access Server	Java SDK	Network Service	TCP/IP
NetServer Console	Java SDK	Network Service	TCP/IP
Transaction Switch	Java SDK	Network Service	TCP/IP

Daemons are not used by any of the components.

Important! Use of the mod_security component that is provided together with Application Server-based on components is mandatory since it comes with a

set of security rules and is an essential part of the application with regard to security (see the document "Administering WAY4™ Application Server" for instructions on setting up the mod_security component).

Note that GigaSpaces XAP needs a license file from the WAY4 vendor. This file must be copied into the "gigaspace" folder of the GS Boostrap application instance. Then the GS Boostrap application must be restarted and WAY4Grid must be deployed.

Important! Automatic creation of JVM Heap Dump must be prohibited for Application Server. This is the default behavior, however it is necessary to ensure that the jvm_heap_dump parameter value is either False or no (see the document "Administering WAY4™ Application Server"):

```
jvm_heap_dump = False
```

Database server requirements

All unused, as well as potentially dangerous operating system services and applications on the server must be stopped or blocked.

Minimum operating system requirements for installing Oracle 12 are described in the following public resources:

- Solaris - https://docs.oracle.com/database/121/SSDBI/pre_install.htm#SSDBI7564
- AIX - https://docs.oracle.com/database/121/AXDBI/pre_install.htm#AXDBI7599
- Oracle Linux 7 and Red Hat Enterprise Linux - https://docs.oracle.com/database/121/LADBI/pre_install.htm#LADBI80757

WAY4 does not require any additional server components other than the "standard version" on AIX 7.2.

WAY4 does not require any additional server components other than the "minimum" package for RHEL 7.2/6.8 and the unzip package.

WAY4 does not require any additional server components other than the "standard version" on Solaris 11.

All unused protocols/services/software on RHEL 7.2/6.8 (for example, IPv6, rpc.statd, rpcbind, cups, postfix, wpa_supplicant, abrtd, certmonger, pulseaudio, etc.), AIX, Solaris and Windows must be disabled.

Memory swapping must be disabled on the database server.

File server requirements

All unused, as well as potentially dangerous operating system services (in particular Windows Restore Points) and applications on the server must be stopped or blocked.

User workstation requirements

All unused, as well as potentially dangerous operating system services (in particular Windows Restore Points) and third-party applications must be stopped or blocked.

Data access requirements

For user access to data in the database, only applications provided with WAY4 can be used, as these applications guarantee audit of user actions in system logs.

The use of third-party applications to access data in the database is strictly prohibited.

NetServer and Transaction Switch

NetServer and Transaction Switch are unattended applications that use a limited set of strictly predefined database queries Communication between the payment application and the database is encrypted by standard DBMS technology. Encryption must be enabled when the payment application is installed. For information, refer to the document "Oracle Database Security and the Payment Card Industry Data Security Standard".

System configuration requirements

To comply with security requirements, the system configuration must meet the following conditions:

1. Access to forms with information about card and account contracts and cardholders is restricted and only granted on a need-to-know basis.
2. Custom elements of the configuration (custom tables, procedures, forms, and pipes) must only store bankcard data in locations for this purpose – in databse tables and special file locations (see the section "Privileges of Access to Standard WAY4 Directories" of the document "Administering Users in WAY4").

Chapter 9. Secure network infrastructure

This chapter addresses PA-DSS Requirement 9.1, shown below.

PA-DSS Requirement	PA-DSS Topic
9.1	Store cardholder data only on servers not connected to the Internet

Any system with payment application components must be located in the bank's internal network, segregated from the demilitarized zone (DMZ).

Database server requirements

The database server must be in a separate segment of the bank's internal network, access to which is protected by a separate firewall.

File server requirements

The file server must be in a separate segment of the bank's internal network, access to which is protected by a separate firewall.

User workstation requirements

Workstations can only access the external network through firewalls.

Data storage requirements

Storage of cardholder information on computers with Internet access is prohibited.

Chapter 10. Remote access

This chapter addresses PA-DSS Requirements 10.1, 10.2.1, and 10.2.3 shown below.

PA-DSS Requirement	PA-DSS Topic
10.1	Implement multi-factor authentication for all remote access to payment application that originates from outside the customer environment.
10.2.1	Securely deliver remote payment application updates.
10.2.3	Securely implement remote-access software.

Multi-factor authentication

Multi-factor authentication is commonly used instead of simple user authentication, where simple authentication is a process during which the initiator of a request provides the other party evidence that he/she is actually who he/she claims to be. Multi-factor authentication is intended to reduce the likelihood that the requestor is providing false evidence of his/her identity. A large number of factors ensure a higher probability that the person providing identification is actually who he/she claims to be in another realm (for example computer system vs. real life). In fact, a larger number of variables should be considered when setting the relative assurance of authentication, than simply the number of "factors" used.

Multi-factor authentication requires that two of the three approved authentication methods be used for authentication. These are the following factors:

- Something the user knows, such as a password or PIN.
- Something the user has, such as a USB token or smart card.
- Something the user is, such as a biometric like fingerprints.

Multi-factor authentication must be used for any remote access to the system.

Non-console administrative access using a USB token

The USB token must be installed and set up according to the manufacturer's instructions. PuTTY with PKCS#11 support must also be installed and configured.

To configure PuTTY for a USB token, use the "PKCS11" tab (these parameters are also used by the SSH agent):

- "Attempt PKCS#11 smartcard auth (SSH-2)" – this parameter is used to enable smart card authentication in general.
- "PKCS#11 library for authentication" – specify the library (DLL) necessary for access to the smart card (PKCS # 11 library files, token labels and certificate labels corresponding to the PKCS#11 middleware, for example, C:\Windows\System32\eToken.dll).

- "Token label" – specify the name of the smart card. This is the same name usually shown when prompted to enter a password when accessing the smart card for cryptographic operations such as signing e-mail.
- "Certificate label" – label for the certificate of the corresponding private and public key to be used for authentication.
- "SSH KeyString" – save the public key to the <Home>/SSH file /authorized_keys on the server.

Connection to Oracle under PMO, OWS_A, OWS_N, SYS, SYSTEM is only allowed from a server and proxy host. The "ON DATABASE LOGIN" trigger controls connections. The trigger analyzes the client's IP address from which the request was received.

For example:

```
create or replace trigger DBA_LOGON
after logon on DATABASE
declare
  v_session V$SESSION%rowtype;
  procedure KILL_SESSION_JOB (p_sid integer, p_serial integer) as
  pragma autonomous_transaction;
    v_job integer;
  begin
    dbms_scheduler.CREATE_JOB(
        job_name => 'JOB_KILL_SESSION_'||p_sid
      , job_type => 'PLSQL_BLOCK'
      , job_action => 'begin execute immediate ''alter system disconnect
session '''''||p_sid||', '||p_serial||''''' immediate''; end;'
      , start_date => sysdate
      , enabled => true
    );
    commit;
  end;
begin
  if sys_context('userenv', 'sessionid') != 0 then
    select s.*
    into v_session
    from V$SESSION s
    where 1=1
      and s.sid = (select sid from V$MYSTAT where rownum = 1);
    if user in ('OWS', 'OWS_A', 'OWS_N', 'SYS', 'SYSTEM') then
      if sys_context('userenv', 'ip_address') is not null and
         sys_context('userenv', 'ip_address') not in ('<oracle-host>',
'<proxy-host>')
```

```
      then
        KILL_SESSION_JOB(v_session.sid, v_session.serial#);
        raise_application_error(-20000, 'Connections by administrator is
allowed only from specific hosts.');
      end if;
    end if;
  end if;
end;
/
```

To connect to a client workstation under PMO, OWS_A, OWS_N, SYS, SYSTEM, on the workstation it is necessary to open an SSH connection from the proxy host (configure client port forwarding to Oracle Listener).

Example (*nix):

```
ssh -N -L <local-port>:<oracle-host>:<oracle-port> <oracle-user>@<proxy-
host>
```

Example (Windows):

```
plink.exe -N -L <local-port>:<oracle-host>:<oracle-port> <oracle-
user>@<proxy-host>
```

On the workstation, the name of the tns using the local port is specified in tnsnames.ora. Example:

```
LOCAL=(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=127.0.0.1)(PORT=<local-
port>))(CONNECT_DATA=(SID=<oracle-sid>)))
```

NetServer

The NetServer application uses two-factor authentication based on user and payment application certificates as the first factor and user password as the second factor.

In addition to two-factor authentication used in NetServer, two-factor authentication must also be used for remote access to the operating system of the server on which the payment application is installed. Recommended technologies: SSH with RSA encryption, TLS and VPN with access certificates. On remote access channels, data must be encrypted using the following encryption technologies: SSH, TLS 1.2 and VPN.

Health Monitoring

To comply with security requirements, it is necessary to do the following:

- Configure "monitoring_ui" and "monitoring" system applications using a secure connection over SSL:
  - Configure "monitoring_ui" and "monitoring" application parameters (see the section "Configuring the "monitoring_ui" System Application" of the document "Administering WAY4 Health Monitoring Gen2"):
    - remoteServiceAPI="ssl".

- rmi_ssl_protocols=TLSv1.2.
- rmi_ssl_cipher_suites, for example, the value TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384.

Examples of parameter values in the "monitoring" application's "config.properties" file:

```
remoteServiceAPI=ssl
keyStore=conf/monitoring-app.jks
keyStorePassword=plain:eyA1xRbh
trustStore=conf/monitoring-app-trust.jks
trustStorePassword=plain:eyA1xRbh
rmi_api_port = 1099
rmi_service_port = 1098
rmi_ssl_protocols=TLSv1.2
rmi_ssl_cipher_suites=TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
```

Examples of parameter values in the "monitoring_ui" application's "config.properties" file:

```
keyStore=conf/monitoring-ui.jks
keyStorePassword=plain:eyA1xRbh
trustStore=conf/monitoring-ui-trust.jks
trustStorePassword=plain:eyA1xRbh
```

- Create certificates for the "monitoring_ui" and "monitoring" applications using the "monitoring" console utility (for MS Windows – "monitoring.bat"), located in the "<AppServer_HOME>/applications/monitoring_ui/app/bin" and "<AppServer_HOME>/applications/monitoring/app/bin" directories:

```
monitoring certificate -g
```

- Exchange (import) certificates must be executed:
  - For Linux:

```
monitoring certificate -e way4@appsrv_host
```

Where way4@appsrv_host – <user@host>.

- For Windows (using the "keytool" utility from the "<AppServer_HOME>/jdk/current/bin" directory):

```
keytool -import -v -trustcacerts -alias monitoring-app -file monitoring-
app.cer -keystore monitoring-ui-trust.jks  -keypass eyA1xRbh -storepass
eyA1xRbh
keytool -import -v -trustcacerts -alias monitoring-ui -file monitoring-
ui.cer -keystore monitoring-app-trust.jks  -keypass eyA1xRbh -storepass
eyA1xRbh
```

- Set up secure access to the "monitoring_ui" utility and add users who will work with this application, see the section "Enabling Two-Factor Authentication" of the document "Administering WAY4™ Application

Server". When adding a user, the "hm_administrator" value must be used for the "role_name" parameter.

WAY4 Web (WS Runtime and Application Server)

For Application Server applications, authorization of applications (and WS API) using client TLS certificates must be enabled, see the section "Authorising Applications using Client Certificates" of the document "Administering WAY4™ Application Server".

To encrypt the connection from the client browser to the IIS web server, configure the WAY4 Web IIS site (web.config configuration file) – the requireSSL attribute value must be "true":

```
...
<configuration>
...
<system.web>
...
  <authentication mode="Forms">
   <forms requireSSL="true">
  </authentication>
...
```

Between the WAY4 Web IIS site and WS Runtime, encryption of the connection to Application Server applications must be set up – ensure that in the web.config file in the WsEngineURL parameter the path to WSR starts with "https://":

```
...
<configuration>
...
 <configSections>
...
  <appSettings>
  <add key="WsEngineURL" value="https://padss-ws:8443/wsruntime_XX_X/ws/"
/>
...
```

Remote access to the file server

Only secure remote access protocols with no identified vulnerabilities at the time of use may be used for access to the file server.

Requirements for remote user workstations

Network communication between remote workstations and the bank's internal network is only possible on wired communications channels ad using a secure

connection (VPN or TLS 1.2), with no identified vulnerabilities at the time of use.

Use of fixed MAC and IP addresses is recommended for access to remote workstations.

Chapter 11. Protection of card data when transmitting over public networks

This chapter addresses PA-DSS requirements 11.1 and 11.2 shown below.

PA-DSS Requirement	PA-DSS Topic
11.1	Secure transmissions of cardholder data over public networks
11.2	Encrypt cardholder data sent over end-user messaging technologies.

Examples of public networks in the scope of PA-DSS:

1. Internet
3. Wireless networks
4. GSM
5. GPRS

Encryption and secure protocols for transmitting data

WAY4 facilitates the sending of PANs by online and end-user messaging technologies. Default setup of WAY4 and standard message/report templates assumes the use of masked PANs. If the unmasked PAN is used in custom templates of end-user messages or reports, encryption with strong cryptography must be applied for each message being sent:

- 3DES with at least a double-length key (112 unpredictable bits)
- AES (128 unpredictable bits)
- RSA with a key of at least 2048 bits.

Important! When the TLS 1.2 protocol is used, RC4 encryption must be switched off. This setting can be enabled using the IIS Crypto utility (see https://www.nartac.com/Products/IISCrypto). Procedure for using the utility:

- In the "Schannel" tab, click on the [Best Practices] button, disable TLS 1.0 and 1.1, leaving only TLS 1.2, and click on the [Apply] button.
- In the "Cipher suites" tab, click on the [Best Practices], disable SHA1, leaving only SHA256 and SHA384, and click on the [Apply] button.
- Restart (this procedure corresponds to version 2.0 of the utility).

Use of encryption when generating reports and end-user messages containing a full, unmasked card number is mandatory for compliance with PCI DSS.

Information about account and card contracts, clients, and other critically sensitive information (including troubleshooting data) can only be transmitted in encrypted form. External devices must be used for data encryption.

Transmission of PIN blocks in any of the supported interfaces is only possible when encrypted with at least a double-length DES key (112 bits).

These requirements must also be considered when developing interfaces to WAY4 that meet a user's specific needs. In particular, when information about a cardholder's contract must be sent using Letters, it is necessary to contact the WAY4 vendor in order for the vendor to ensure secure (encrypted) storage of letter templates in the database and secure data transmission.

The table below contains information about WAY4 product and component compliance with requirements for transmitting sensitive data.

Name of product or component	**Compliance**
Remote access	VPN must be used to transmit data. **Important!** Not using VPN will result in noncompliance with PCI DSS.
e-Commerce issuing, e-Commerce acquiring, Bill payments	3-D Secure transmits data on secure channels (TLS 1.2).
ATM	VPN must be used to transmit data. **Important!** Not using VPN will result in noncompliance with PCI DSS.
POS	3DES with at least a double-length key (112 bit)
H2H	VPN must be used to transmit data. **Important!** Not using VPN will result in noncompliance with PCI DSS.
Transaction Switch based e-Commerce	3DES with at least a double-length key (112 bit)
Lifestyle Banking	All cardholder data must be transmitted over public networks using the HTTPS protocol. The HTTPS protocol must be configured to use TLS 1.2.
Way4 Rainbow Web Banking, Mobile Web Banking, Customer Profile and Telegram	All cardholder data must be transmitted over public networks using the HTTPS protocol. The HTTPS protocol must be configured to use TLS 1.2.
HCE	AES, 128 bits

Masking PAN

When sending a PAN on a public network without encryption, it must be masked according to PCI DSS requirements.

This requirement must also be considered when developing custom interfaces to WAY4.

WAY4 SMS Banking masks PAN in outgoing SMS messages.

Chapter 12. Encryption of non-console administrative access

This chapter addresses PA-DSS Requirements 12.1 and 12.2 shown below.

PA-DSS Requirement	PA-DSS Topic
12.1	Encrypt non-console administrative access.
12.2	Use multi-factor authentication for all personnel with non-console administrative access.

To encrypt non-console administrative access to the system, secure protocols must be used that have no identified vulnerabilities at the time of use.

Currently, these protocols are:

- SSH
- VPN
- TLS 1.2

Use of secure protocols for non-console administrative access is mandatory for compliance with PCI DSS.

Instructions for setting up non-console administrative access are provided in the document "Secure Access to the Oracle Database in Compliance with PCI DSS".

NetServer and Transaction Switch

Multifactor authentication must be used to access the operating system of the server on which the payment application is installed. Recommended technologies: SSH with RSA encryption, TLS 1.2 and VPN with access certificates. On remote access channels, data must be encrypted using the following encryption technologies: SSH, TLS 1.2 and VPN.

Chapter 13. Testing

System testing data requirements

Real data cannot be used for testing.

If the test system is created from a production system, account and card contract data, client data, key values and other sensitive data must first be obfuscated.

Chapter 14. Hardware Security Module (HSM) Setup

The following settings are recommended for Thales hardware security modules (HSM):

- Disable the "P0" host command (Host Command). The command is disabled through the HSM console; instructions for disabling it are provided in the appropriate HSM documentation.

  The "P0" command must be disabled only for HSMs that are not used in the electric personalization module.

- Use a separate HSM for personalization of smart cards.
- When executing the "CS" (Configure Security) console command, it is recommended to specify the following answers to questions:
  - Echo [oN/ofF]: **F**
  - Select clear PINs? [Y/N]: **N**
  - Enable Single-DES? [Y/N]: **N**
  - Prevent Single-DES keys masquerading as double or triple-length key? [Y/N]: **Y**
  - Single/double length ZMKs [S/D]: **D**
  - Restrict Key Check Values to 6 hex chars [Y/N]: **Y**
  - Enable multiple authorised activities [Y/N]: **Y**
  - Enable PIN Block Format 34 as output format for PIN Translations to ZPK [Y/N]: **N**

    If the "PIN Change" option is used, specify "**Y**" as the response.
  - Key export and import in trusted format only? [Y/N]: **N**

Appendix 1. Sample Key Custodian Form

This document is an example Key Custodian form for key custodians to acknowledge that they understand and accept their key-custodian responsibilities. Any user who has access to any encryption keys used in conjunction with the payment application must agree and sign a document such as this.

A key custodian is responsible for maintaining the confidentiality and integrity of keys in their custody. A key custodian must protect access to all encryption keys in their custody.

I, _____, as an employee of _____ hereby agree that I:

1) Have read and understood the policies and procedures associated with key management and agree to comply with them to the best of my ability.

2) Agree to never compromise the security of the keys in my custody by divulging any information about key management practices, related security systems, passwords, or other private information associated with the company's systems to any unauthorized persons.

3) Agree to immediately report any suspicious activity that may compromise key security

Printed Name: _____

Title: _____

Date: _____

Signature: _____