IBM Counter Fraud Management for Safer Payments 5.6.0
Version 5 Release 6

Implementation Guide

SC34-2787-00

IBM Counter Fraud Management for Safer Payments 5.6.0
Version 5 Release 6

Implementation Guide

SC34-2787-00

This edition applies to Version 5 Release 6 of IBM Counter Fraud Management for Safer Payments, Program Number 5725-Z82, and to all subsequent releases and modifications until otherwise indicated in new editions.

A form for readers' comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Deutschland Research & Development GmbH
Department 3282
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

You may also send your comments by FAX or via the Internet:

Internet: s390id@de.ibm.com
FAX (Germany): 07031-16-3456
FAX (other countries): (+49)+7031-16-3456

When you send information to IBM®, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

About this publication

This publication describes how to install, configure, and operate IBM Counter Fraud Management for Safer Payments 5.6.0.*x*.

This Implementation Guide is valid for Safer Payments 5.6.0.*x*. See "Versioning Method" on page x for more information about Safer Payments version numbers.

To obtain the most current version of the Safer Payments Implementation Guide, go to the IBM Support Portal or request a copy from your account manager.

Note: Subsequently, IBM Counter Fraud Management for Safer Payments is simply referred to as "Safer Payments".

PA-DSS information

Important: IBM Counter Fraud Management for Safer Payments 5.6.0 is not certified against PA-DSS. This document describes how to implement Safer Payments in a PCI DSS compliant environment based on our experience with the certified predecessor version 5.5.

Who should use this publication

This publication is intended as a reference for system administrators and Safer Payments administrators who deploy Safer Payments in a PCI DSS compliant environment.

How to use this publication

The publication consists of the following sections:

- Chapter 1, "Installing and configuring Safer Payments," on page 1 describes how to install, configure, and operate Safer Payments.
- Chapter 2, "Key management procedures for cryptographic keys," on page 45 describes the cryptographic keys that are used by Safer Payments, how keys are generated, and how to enter and activate keys.

Where to find more information

IBM Counter Fraud Management for Safer Payments Home Page

IBM Counter Fraud Management for Safer Payments has a home page on the World Wide Web, which offers up-to-date information about related products and services, new functions, and other items of interest.

You can find the IBM Counter Fraud Management for Safer Payments home page at:

https://www.ibm.com/marketplace/payment-fraud-prevention

The IBM Support Portal is the central hub for support including Technotes:

https://www.ibm.com/support/entry/portal/support

Fix Central provides fixes and updates for IBM Counter Fraud Management for Safer Payments:

https://www.ibm.com/support/fixcentral

Versioning Method

IBM Counter Fraud Management for Safer Payments uses a release number scheme that is based on the PA-DSS versioning recommendations. The versioning of IBM Counter Fraud Management for Safer Payments consists of four numbers that are separated by dots. For example, 5.5.0.0.

- The first number denotes the project generation.
- The second number denotes feature releases or high impact changes per PA-DSS.
- The third number denotes low impact changes per PA-DSS.
- The fourth number denotes no impact changes. These can also be denoted with a wildcard ("*x*") For example, 5.5.0.*x*.

Changes in the fourth number denote changes, which are not PCI relevant and therefore have no impact per PA-DSS Program Guide.

The following definitions are taken from the PA-DSS Program Guide.

High Impact Changes

High impact changes to the payment application where any of the following apply:

- Four or more PA-DSS Requirements are affected, not including Requirements 13 and 14.
- Half or more of all PA-DSS Requirements/sub-Requirements are affected, not including Requirements 13 and 14.
- Half or more of the payment application's functionality or half or more of its code-base is changed.
- Addition of tested platform/operating system to include on the list of validated payment applications.

High Impact changes require the vendor to submit the new version of the payment application for a full PA-DSS assessment.

See *Section 5.2.3.4 "High Impact Changes"* in the PA-DSS Program Guide for details.

Low Impact Changes

Low impact changes to the payment application where all of the following conditions are met:

- Three or fewer PA-DSS Requirements are affected, not including Requirements 13 and 14.
- Less than half of all PA-DSS Requirements/sub-Requirements are affected, not including Requirements 13 and 14.
- Less than half the Payment Application's functionality is affected and less than half the Payment Application's code-base is changed.

Low Impact changes might be eligible for partial or "delta" assessment.

See *Section 5.2.3.3 "Low Impact Changes"* in the PA-DSS Program Guide for details.

No Impact Non-security-related changes

No Impact Non-security-related changes that have no impact to PA-DSS related functions, tested platforms, operating systems, or dependencies and no impact on any of the PA-DSS Requirements.

No Impact changes might be eligible for partial or “delta” assessment. In addition, no impact non-security-related changes are eligible for wildcard changes to the version number. Wildcard changes do not result in a change of the application listing on the PCI Council website.

See *Section 5.2.3.2 "No Impact Changes"* in the PA-DSS Program Guide for details.

Administrative Changes

Administrative changes to the payment application listing or changes to how the payment application is described in the list of validated payment applications, for example, corporate identity or application name changes.

See *Section 5.2.3.1 "Administrative Changes"* in the PA-DSS Program Guide for details.

Chapter 1. Installing and configuring Safer Payments

This section describes how to install and configure Safer Payments so that it meets all PCI DSS requirements. Further sections address software updates, ongoing operation of Safer Payments, and it's decommissioning.

Safer Payments provides a built-in PCI DSS compliance report that lists all relevant configuration settings that must be changed to achieve PCI DSS compliance. See "PCI DSS compliance report" on page 43 for details.

Preliminary considerations

Before you install and configure Safer Payments, your organization needs to define and implement certain operational processes, and periods.

Define and implement operational processes

To achieve PCI DSS compliance, it is not enough to configure Safer Payments as described here. You must also implement a set of operational processes within your organization for PCI DSS compliant operation.

Note: Therefore, it is important that you read the PCI DSS documentation and implement the operational processes that are described there.

https://www.pcisecuritystandards.org/document_library

Define a cryptoperiod

The cryptoperiod defines the lifetime of an encryption key. At the end of each cryptoperiod, keys must be replaced.

PCI DSS itself does not postulate a specific cryptoperiod. However, it is necessary that you as an organization define your own cryptoperiod. See "Enforce regular key changes" on page 52 for details.

Define a retention period

Outdated cardholder data must be securely deleted. PCI DSS itself does not postulate when cardholder data becomes outdated. However, according to PCI DSS requirement 3.1 (aligns with PA-DSS requirement 2.1) it is necessary that you as an organization define a retention period.

You can define different retention periods for different kind of data elements:

- A retention period for transaction data, according to your business requirements.
- A longer retention period for all other data, such as cases, or event logs.

Basically, you could also define the same retention period for both types of data. Retention requirements for cases or audit trails are typically longer than five years. However, usually there is no business need to retain transaction data for such extended periods, and memory consumption would be heavy given the typical transaction volumes.

Installation

This section describes how to install, uninstall, and update Safer Payments.

The instructions assume that you install Safer Payments as a cluster of multiple Safer Payments instances. If you want to install Safer Payments as stand-alone service, omit the installation steps for the other instances.

You must be logged in with an administrator account on your workstation to run some of the installation steps described.

Note: You do not need administrator privileges to run Safer Payments.

System requirements

This section lists the currently supported operating systems and further requirements.

Supported platforms

Safer Payments is tested for the following operating systems:

- Red Hat Enterprise Linux 6 (RHEL 6)
- Red Hat Enterprise Linux 7 (RHEL 7)
- Oracle Linux 7

User access is provided with all recent standard browsers. The following browsers are fully tested for compatibility:

- Internet Explorer 10 or later
- Firefox 4 or later
- Google Chrome 24 or later

Apple Safari is partially tested for compatibility.

Screen resolution

Safer Payments screen pages are designed to work with a minimum resolution of 1280×768 pixel, thus WXGA (1280×768) screen resolution is the minimum for correct page display. For power users of Safer Payments, for example, fraud analysts, a dual monitor configuration with HDTV (1920x1080) screen resolution is recommended. With this resolution users can open multiple browser pages/tabs with the same Safer Payments session. Because of the high interactiveness of the Safer Payments user interface, the performance of the JavaScript engine is key to smooth operation of Safer Payments. The Google Chrome browser provides the best user interface performance for Safer Payments.

Download and verify installation image

You can obtain the setup file `SaferPayments_5_5_x_x.zip` from Passport Advantage.

To verify that the installation image is not corrupted, you must check its integrity with an SHA256 checksum checker. Your IBM representative can provide you with the correct checksum of the installation image.

In Linux/Unix operating systems, use the preinstalled sha256sum tool to verify that the checksum provided by the tool matches with the checksum provided by IBM. To run checksum, enter:

```
sha256sum SaferPayments_5_5_x_x.zip
```

Note: If you are using a Windows operating system, you can run the SHA256 Checksum Utility. It is available from https://kanguru.zendesk.com/entries/21747773-SHA256-Checksum-Utility.

After you have run checksum, you can extract the .zip file to a temporary directory. The .zip file contains the following files:

SaferPayments.bin
The installation image file.

ibm_jre_8.0.2.10_linux_x64.vm
The setup file to install LINUX IBM JRE 1.8 SR2 FP10 (64-bit). You need this file only, if you don't have JVM installed.

installer.properties
The sample response file. You need this file only, if you want to run a silent installation.

Run the installer to extract the installation image

This topic describes how to extract the installation image with the installer.

To run the installer, Java Runtime Environment (JRE), or Java Development Kit (JDK) must be installed on your system. If you don't have JRE or JDK installed, you can use the `ibm_jre_8.0.2.10_linux_x64.vm` file that is contained in the setup file to install LINUX IBM JRE 1.8 SR2 FP10 (64-bit).

Important: The Linux IBM JRE is intended for use only with InstallAnywhere installers. Do not use it for any other purposes.

Root privileges are not needed to use Linux IBM JRE. Log in on the console and run the following command:

```
unzip ibm_jre_8.0.2.10_linux_x64.vm
tar xf vm.tar.Z
chmod +x jre/bin/java
chmod +x SaferPayments.bin
su
export PATH=$PATH:`pwd`/jre/bin
```

Note: You must set the **export** command with the same user as the install user.

Initial installation

This topic describes how to install Safer Payments if you don't have a previous Safer Payments installation.

Preinstallation tasks

Before you start the initial installation, you must create:

SPUser

SPUser is the user, which runs the Safer Payments instance.

SPUserGroup

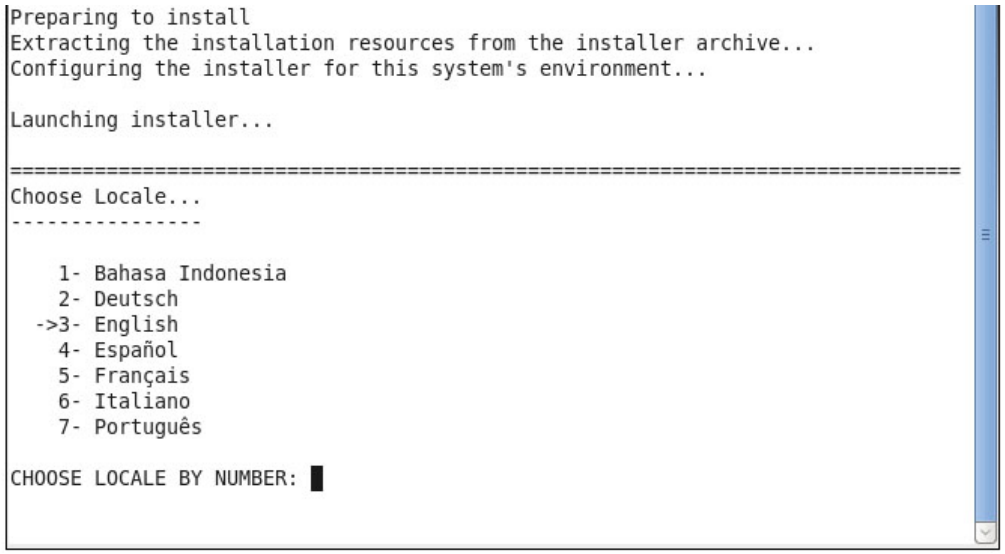
SPUser belongs to the SPUserGroup.

Installation steps

Log in as root on the console and run the following command:

```
sh ./SaferPayments.bin
```

The installer starts in console mode:



Select a language, press the enter key, and follow the installation steps.

In the installer you can specify some special settings:

Change license language
Even if you selected a language other than English, the English license is shown by default. To display the license in the selected language press 5 when the English license is shown.

Choose install set

Install Set Typical
If you choose `Install Set Typical` the configuration with the type "empty" is copied to the factory reset folder. The configuration folder must be created manually after installation.

Install Set Custom
If you choose `Install Set Custom` the configuration folder is created automatically and an option is provided to select to most appropriate.

Install Set Custom settings

Configuration
This is the Safer Payments configuration that you want to start with in your configuration folder.

Path The Safer Payments home directory, where you want to configure your Safer Payments installation. The configuration is copied to this path.

SPUser/SPuserGroup
The user and group that are created in "Preinstallation tasks" on page 3.

The following folders are created:

/usr/bin
The Safer Payments and the Keygen binary files are installed in this folder.

/usr/lib64
The AES and SQL libraries of Safer Payments are located in this folder.

/opt/ibm/safer_payments/install
The Safer Payments default installation directory.

/opt/ibm/safer_payments/install/inc
The JavaScript files of Safer Payments are located in this folder.

/opt/ibm/safer_payments/install/factory_reset
This folder contains an initial configuration to start Safer Payments for the first time. Never change the files in this folder and always use a copy with other user privileges for your initial configuration.

Silent Installation

In silent mode, the installer has no user interaction and is run by using a response file that contains the values for various variables.

Safer Payments provides a sample response file that is called `installer.properties` with default values. To accept the license agreement, open the sample response file and set:

```
$LICENSE_ACCEPTED$=true
```

Make sure the response file and `SaferPayments.bin` are in the same directory.

Log in as root on the console and run the following command:

```
sh ./SaferPayments.bin -i silent
```

Postinstallation

If you choose `Install Set Typical` you must run the following commands after installation:

```
cp -R /opt/ibm/safer_payments/install/factory_reset/* /path
```

```
chown -R SPUser:SPUserGroup /path
```

- */path* is the Safer Payments home directory, where you installed your Safer Payments installation.
- *SPUser* is the user, which runs the Safer Payments instance, with *SPUser* belonging to the *SPUserGroup* as defined in "Preinstallation tasks" on page 3.

Uninstallation

This topic describes how to uninstall Safer Payments. Uninstalling Safer Payments uninstalls only installation files and folders, not configuration data.

To uninstall, navigate to:

`/path/IBMSafer Payments_installation`

Log in as root on the console and run the following command:

```
./uninstall_safer_payments
```

Silent Uninstallation

In silent mode, the uninstaller has no user interaction and is run by using a response file that contains the values for various variables.

To run the silent uninstaller, navigate to:

```
/path/IBMSafer Payments_installation
```

Log in as root on the console and run the following command:

```
./uninstall_safer_payments -i silent
```

Feature update

A feature update is indicated by a change of the second or third revision number position.

In a feature update, file formats might be changed so that you cannot change back to an earlier release. Also, you might not be able to install such an update immediately, that is, on a running Safer Payments cluster that still fully runs during update. There might be specific instructions for the type of feature update you plan.

Depending on your specific application needs, it might be advisable to contact IBM support for assistance with a feature update.

Read the release notes to understand if any specific tasks are necessary to install the update.

Shut down all Safer Payments instances in a cluster and back up all Safer Payments instances to revert the update in case it fails fatally.

If you have a previous rpm installation, you must uninstall it before you run the installer. Log in as root on the console and run the following command:

```
rpm -e iris
```

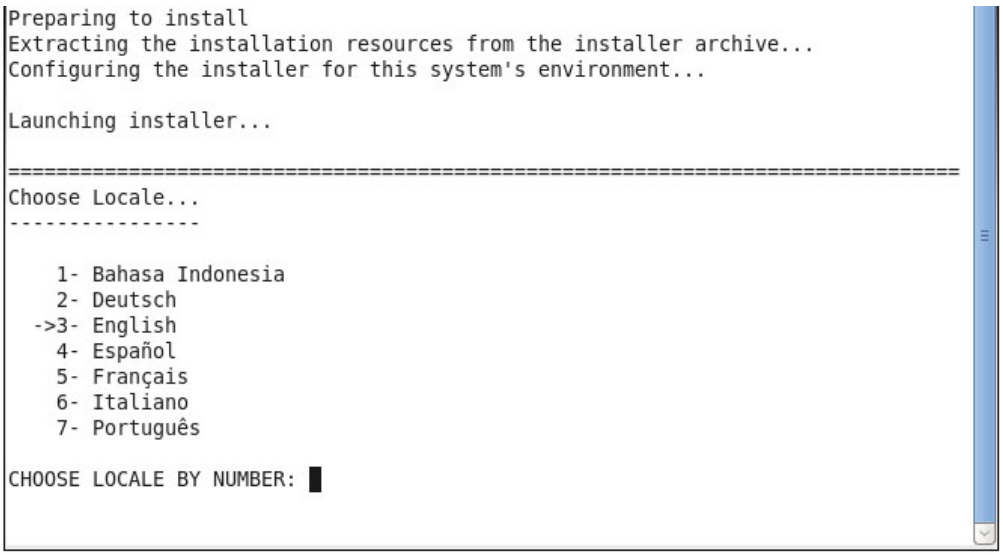
To run the installer, Java Runtime Environment (JRE), or Java Development Kit (JDK) must be installed on your system. If you don't have JRE or JDK installed, you can use the `ibm_jre_8.0.2.10_linux_x64.vm` file that is contained in the setup file to install LINUX IBM JRE 1.8 SR2 FP 10 (64-bit).

Important: The Linux IBM JRE is intended for use only with InstallAnywhere installers. Do not use it for any other purposes.

Log in as root on the console and run the following command:

```
sh ./SaferPayments.bin
```

The installer starts in console mode:



Select a language, press the enter key, and follow the installation steps.

Note: Do not use `Install Set Custom` for feature updates.

The following folders are created:

/usr/bin
The Safer Payments and the Keygen binary files are installed in this folder.

/usr/lib64
The AES and SQL libraries of Safer Payments are located in this folder.

/opt/ibm/safer_payments/install
The Safer Payments default installation directory.

/opt/ibm/safer_payments/install/inc
The JavaScript files of Safer Payments are located in this folder.

/opt/ibm/safer_payments/install/factory_reset
This folder contains an initial configuration to start Safer Payments for the first time. Never change the files in this folder and always use a copy with other user privileges for your initial configuration.

If you have a previous rpm installation, you must update all symbolic links (readme file and swidtag, license, inc folder) of all configurations after the installation. Navigate to configuration folder, log in as `SPUser` on the console and run the following commands:

```
ln -f -s /installationFolder/readme readme
ln -f -s /installationFolder/swidtag swidtag
ln -f -s /installationFolder/license license
ln -f -s /installationFolder/inc inc
```

Read the release notes to understand if there are any additional files that require exchange. Start all updated Safer Payments instances.

Patch update

A patch update is indicated by a change of the fourth revision number position.

If not stated otherwise in the release notes, full exchangeability between patch updates is maintained. That is, you can switch freely back and forth between Safer Payments revisions that share the first three revision numbers. Patch releases contain bug fixes only and all interfaces and file formats remain the same or can easily be converted.

It is not necessary to shut down an instance before applying the installer update. However, you must disable the API and FLI because no instance should have an active API/FLI during the update.

Perform the following steps on each instance:

1. Log in as root on the console and run the following command:
   ```
   sh ./SaferPayments.bin
   ```
   The installer starts in console mode. Select a language and follow the installation steps.

   Note: Do not use `Install Set Custom` for feature updates.
2. Deactivate the MCI and BDI interfaces of the Safer Payments instance to be updated (do not disable the ECI interface) to redirect their data streams to another instance.
3. Ensure that all outgoing queues of the Safer Payments instance to be updated are empty.
4. Next, you must shut down the updated Safer Payments instance and then restart it. You can do this by going to **Cluster** and selecting the instance with a right mouse click or via service script.

   Important: You must shut down the instance. Only restarting is not sufficient.
5. Reactivate the FLI interface of the updated Safer Payments instance.
6. Reactivate the MCI/BDI interface of the updated Safer Payments instance.

Make sure that the FLI is enabled on all instances and wait until all instances are synchronized in the cluster. Re-enable the API.

Note: You can operate Safer Payments instances of different release numbers in parallel for a short time. However, restrict this to the time you need to update all instances in a cluster, as there might not always be the full Safer Payments functionality available when you operate a cluster with different releases. See the release notes for details.

Basic configuration

This section describes the basic configuration steps.

Start the first Safer Payments instance

The browser-based Safer Payments user interface is used to configure a Safer Payments cluster. To access it, you must first start the first Safer Payments cluster instance.

1. To start the first Safer Payments cluster instance, run the following commands from the console on the server:
   ```
   su SPUser
   cd /myInstallation/cfg
   iris id=i createinstances=n
   ```

- */myInstallation* is the folder where you want to install your Safer Payments installation.
- *SPUser* is the user, which runs the Safer Payments instance.
- *i* must be a unique ID of the instance you are currently installing. Preferably, start your first instance with 1. That is, if you set up three instances in total, use IDs 1, 2, and 3.
- *n* is the number of instances you want to create.

2. Check the system event log messages on the console window, and verify that they indicate a proper start of the Safer Payments cluster instance. That is, no warning (W), error (E), or fatal (F) type messages.

   Exception: The `status.iris` file does not exist yet and is being created during the first start. This creates an E 155 during the first start, followed by a message, that the file has been created. Therefore, this error message is expected.
3. Depending on the configuration of the server that you are installing on, you might have to configure the firewall open port, the API port for HTTP access of the browser. The default HTTP port of the Safer Payments first instance is "8001".

   Open a browser and enter:

   `http://127.0.0.1:8001`
4. The login screen of Safer Payments is displayed.
5. Enter **user** as login and **12345678** as password. You are prompted to change the password of this account immediately.

   Note: To comply with PA-DSS requirement 3.1, you must create new personalized users for your configuration and disable the default configuration user.
6. Log in with one of your new users and continue the configuration.
7. The full user interface of Safer Payments is displayed.
8. Click the **Cluster** tab
9. The **Cluster Settings** window shows a table with one row for each Safer Payments instance.

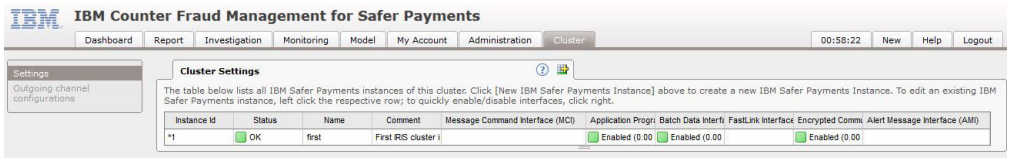


Figure 1. Cluster Settings window

10. Click the equivalent row to open the configuration details of a Safer Payments instance.

Since Safer Payments was started without a previous configuration, it uses default settings for the number of cluster instances you specified with the `createinstances` command.

To use all Safer Payments interfaces, it might be required to open more ports in your firewall. By default Safer Payments uses the following ports:

- 8001 - Application Programming Interface

- 37911 - Message Command Interface
- 37921 - Fast Link Interface
- 37931 - Status Control Interface
- 37941 - Encrypted Communication Interface

Note: If you choose to operate a MQ server to deliver data to Safer Payments, you must set up your firewall appropriately.

Cluster instance configuration

You can now customize all cluster settings. This includes changing the IP addresses/ports, enabling SSL encryption as described in "Configure SSL encryption," limiting IP address ranges, and changing local file storage locations as described in "Configure cardholder data storage locations" on page 16.

Make the appropriate settings for all cluster instances, not only the instance you are currently working on, even if the others are not set up physically yet.

Note: Changes to the local file storage are processed after a restart of a Safer Payments instance. Thus you can move the files while the instance is offline. All changes to the interfaces are processed immediately when the settings are saved.

Configure SSL encryption

Note: TLS is the successor of SSL. Subsequently, the term SSL is used to refer to the secure communication technologies within Safer Payments. In the Safer Payments interfaces all equivalent elements are named SSL.

For PCI DSS compliance you must enable SSL encryption as follows:

- The API must be encrypted to securely transmit passwords.
- The MCI must be encrypted when cardholder data is sent over public networks.
- The ECI must be enabled for synchronization of encryption keys between cluster instances.
- SSL and early TLS are not considered strong cryptography. Payment applications must not use, or support the use of, SSL or early TLS. Therefore, TLS 1.0 and 1.1 must be disabled for API / MCI and ECI.

For each interface that uses SSL encryption, encrypted SSL certificate files must be provided. Safer Payments needs two files to support an encrypted connection. The server certificate and the private key in PEM format. The storage location of these files can be configured in the **SSL Settings** window. See"Create certificates with OpenSSL" on page 13 for details on how to create the required certificates.

1. On the Safer Payments user interface, click the **Cluster** tab.

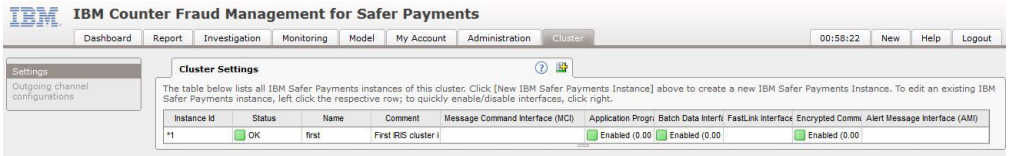


Figure 2. Cluster settings window

2. Click the first instance of the **Cluster Settings** table.

3. Scroll down to the **Application Programming Interface (API)** window.

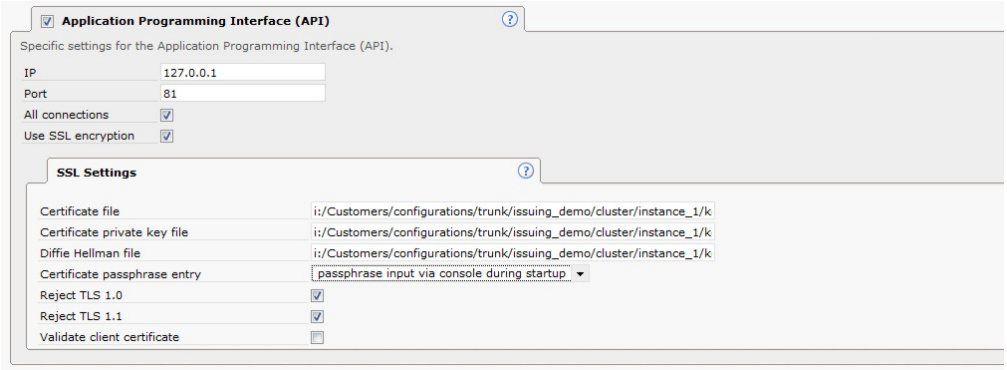


Figure 3. API - SSL settings window

4. Select the **Application Programming Interface**, **Reject TLS 1.0** and **Reject TLS 1.1** check boxes.
5. Scroll down to the **Encrypted Communication Interface (ECI)** window and select the same check boxes.
6. Repeat these steps for each instance.

Note:

- SSL settings are individual for each Safer Payments instance because different instances of Safer Payments running on different computers with different IP addresses require different certificates.
- Enabling SSL encryption and changing the settings becomes effective immediately.
- From now on, you are prompted to enter the certificate passphrase on the console during startup for each instance. See “Start and stop Safer Payments instances” on page 35 for details.

Sending cardholder data over public networks

If cardholder data is to be sent over public networks, Safer Payments must also validate the SSL certificates, and multi-factor authentication is enforced for the API.

If cardholder data is to be sent over public networks, Safer Payments must validate the SSL client certificates for MCI and ECI. Furthermore, API access needs to be secured by multi-factor authentication using either a third-party solution (for example VPN) or by activating the validation of individual API client certificates as the second authentication factor.

Make sure that the “validate client certificate CN” option is enabled as well. This enforces individual certificates for each user, which must use the users login name as common name (CN).

Note: Before activating this option, make sure that you have created proper client certificates for at least the administrative Safer Payments users. The creation of client certificates for this purpose is covered by the section Create certificates with OpenSSL.

To change the API, MCI and ECI settings open the instance configuration for each cluster instance.

1. Click the **Administration** tab.

2. Select **Cluster** from the left navigation pane.
3. Select an instance from the cluster instance table.
4. Scroll down to the **Message Command Interface (MCI)** window and select the **Validate client certificate** check box.
5. Place the client CA certificate file in the `/key/` directory of the instance.

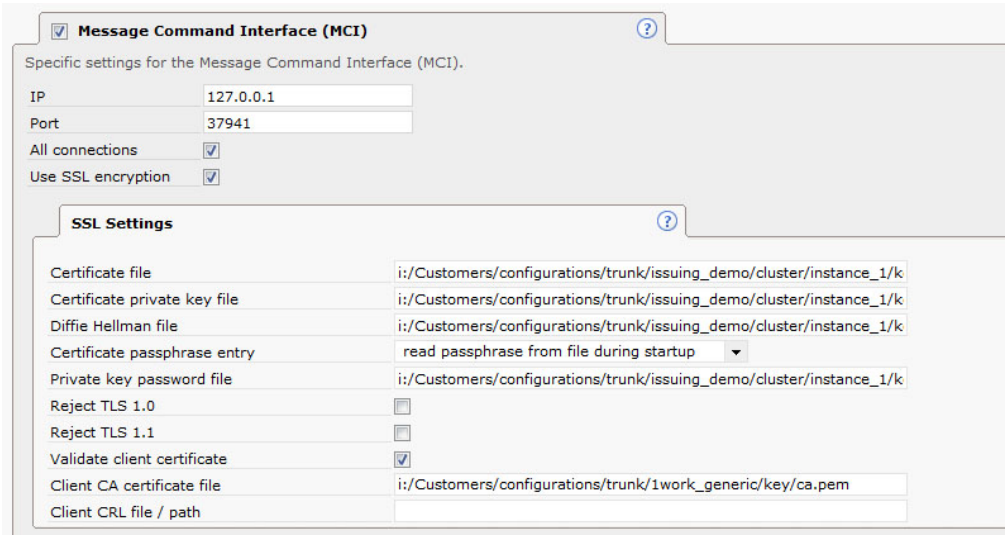


Figure 4. Message Command Interface (MCI) window

6. Scroll down to the **Application Programming Interface (API)** window and select the **Validate client certificate** check box.

   Note: You need a client CA certificate for each Safer Payments instance, and a corresponding certificate for each service consumer that is used to access Safer Payments.
7. Place the client CA certificate file in the `/key/` directory of the instance.

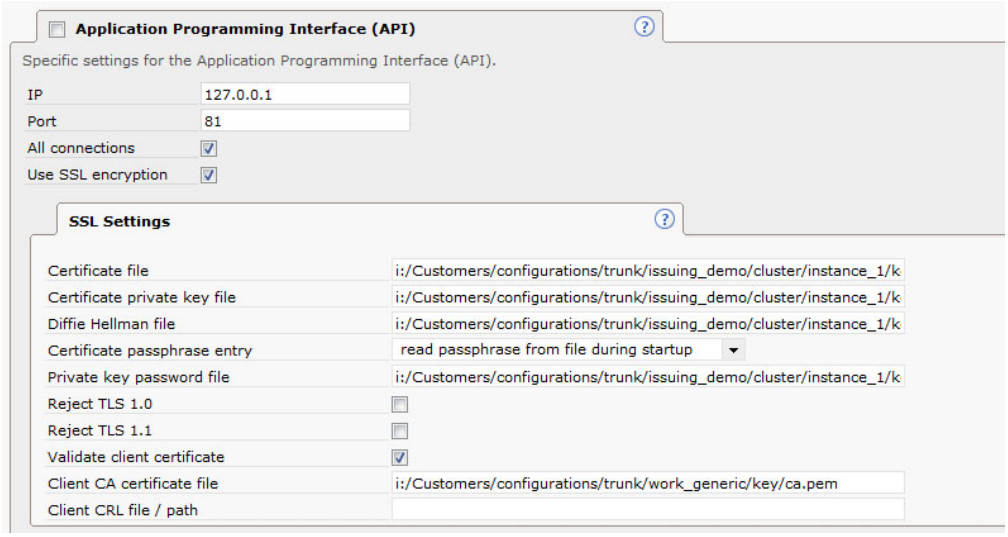


Figure 5. Application Programming Interface (MCI) window

8. Scroll down within the instance settings to the **Encrypted Communication Interface** window.

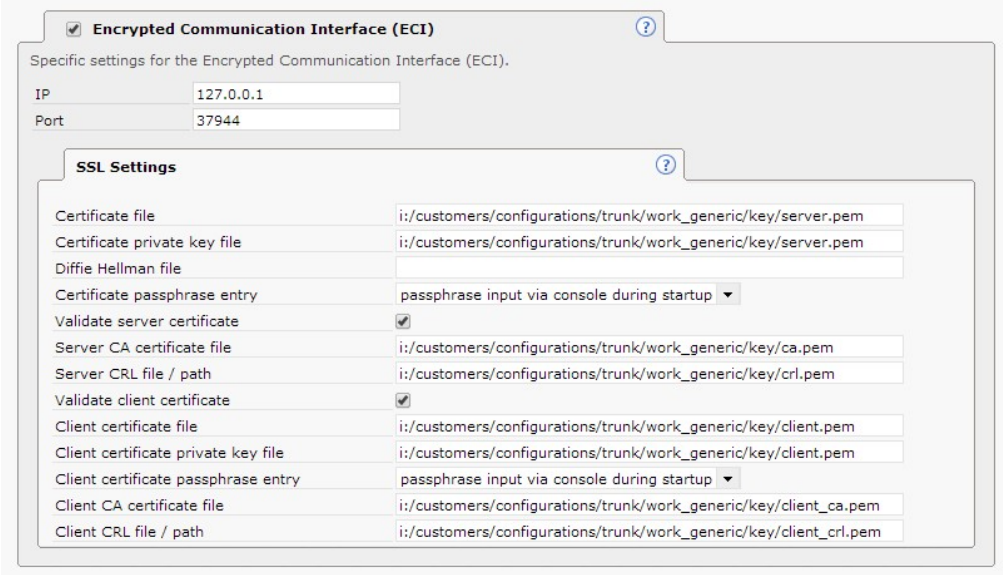


Figure 6. Encrypted Communication Interface (ECI) window

9. Select the **Validate server certificate** and **Validate client certificate** check boxes.

   Note: You need both a server and client CA certificate for each Safer Payments instance, and a corresponding client certificate.

   The encrypted private key is usually stored within the client certificate file but can optionally be stored in a separate file. The **Client certificate private key file** entry points to the correct location.
10. Place the files in the `/key/` directory of the instance.
11. Optionally, **Server CRL file / path** and **Client CRL file / path** can be used to define certificate revocation lists.

Create certificates with OpenSSL

This topic shows the steps that you need to run to create certificates with OpenSSL.

Important: You must ask your security expert to review and run these steps. IBM takes no guarantee for security, as these steps might differ on different platforms.

The steps that are described here are based on: http://codeghar.wordpress.com/2008/03/17/create-a-certificate-authority-and-certificates-with-openssl/

The settings shown here are adapted to Safer Payments.

1. **Create config file**

   The `caconfig.cnf` file is the default config file for the certificate authority (CA). It has the following content:

```
#.................................
[ ca ]
default_ca = CA_default
[ CA_default ]
dir = .
```

```
certs = $dir/certs
crl_dir = $dir/crl
database = $dir/index.txt
new_certs_dir = $dir/newcerts
certificate = $dir/certs/cacert.pem
serial = $dir/serial
crl = $dir/crl/crl.pem
private_key = $dir/private/cakey.pem
#RANDFILE = $dir/private/.rand
x509_extensions = usr_cert
crl_extensions = crl_ext
default_days = 3650
#default_startdate = YYMMDDHHMMSSZ
#default_enddate = YYMMDDHHMMSSZ
default_crl_days = 183
#default_crl_hours = 24
default_md = sha256
preserve = no
#msie_hack
policy = policy_match

[ policy_match ]
countryName = match
#stateOrProvinceName = match
#localityName = match
organizationName = match
commonName = supplied
emailAddress = optional

[ req ]
default_bits = 4096 # Size of keys
default_keyfile = key.pem # name of generated keys
distinguished_name = req_distinguished_name
default_md = sha256 # message digest algorithm
attributes = req_attributes
x509_extensions = v3_ca
#input_password
#output_password
string_mask = nombstr # permitted characters
req_extensions = v3_req

[ req_distinguished_name ]
countryName = Country Name (2 letter code)
countryName_default = DE
countryName_min = 2
countryName_max = 2
#stateOrProvinceName = State or Province Name (full name)
#stateOrProvinceName_default = RLP
#localityName = Locality Name (city, district)
#localityName_default = Coblence
organizationName = Organization Name (company)
organizationName_default = IRIS
organizationalUnitName = Organizational Unit Name (department, division)
organizationalUnitName_default = Fraud Prevention
commonName = Common Name (hostname, IP, or user name)
commonName_max = 64
commonName_default = 192.168.1.1
emailAddress = Email Address
emailAddress_max = 40
emailAddress_default = support@iris.de

[ req_attributes ]
#challengePassword = A challenege password
#challengePassword_min = 4
#challengePassword_max = 20
#unstructuredName = An optional company name
```

```
[ usr_cert ]
basicConstraints= CA:FALSE
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always
#nsComment = ''OpenSSL Generated Certificate''
#nsCertType = client, email, objsign for ''everything including object signing''
subjectAltName=email:copy
issuerAltName=issuer:copy
#nsCaRevocationUrl = http://www.domain.dom/ca-crl.pem
#nsBaseUrl =
#nsRenewalUrl =
#nsCaPolicyUrl =
#nsSslServerName =

[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

[ v3_ca ]
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer:always
basicConstraints = CA:TRUE
#keyUsage = cRLSign, keyCertSign
#nsCertType = sslCA, emailCA
#subjectAltName=email:copy
#issuerAltName=issuer:copy
#obj=DER:02:03

[ crl_ext ]
#issuerAltName=issuer:copy
authorityKeyIdentifier=keyid:always,issuer:always
#.................................
```

2. **Create Diffie-Hellman files**

   ```
   $ openssl dhparam -out dh2048.pem 2048
   ```

3. **Create CA**

   ```
   $ mkdir ~/myca
   $ cd ~/myca
   $ mkdir private certs newcerts conf export crl
   $ echo "01" > serial
   $ touch index.txt
   $ vim conf/caconfig.cnf (Step 1)
   $ openssl req -new -x509 -extensions v3_ca -keyout private/cakey.pem -out
   certs/cacert.pem -days 3650 -config conf/caconfig.cnf
      → PW: xxxxxxxx
   ```

   Note: Create a strong password and distribute it only to entitled people.

4. **Create signed server certificate and private server key**

   You need one certificate/key for each Safer Payments instance. Run the following commands once for each instance and replace SERVER_IP with the IP address or host name of the server.

   ```
   $ openssl req -new -nodes -config conf/caconfig.cnf -out SERVER_IP.req.pem
   -keyout private/SERVER_IP.key.pem
      → CN: SERVER_IP
   $ openssl ca -config conf/caconfig.cnf -out newcerts/SERVER_IP.cert.pem
   -infiles SERVER_IP.req.pem
   ```

5. **Create signed client certificate and private client key**

   For MCI and ECI, you need at least one client certificate for each instance. Run the command twice per instance with unique file names and make sure that you enter unique common names when prompted.

If you want to use multi-factor authentication using API client validation, you might want to create one extra client certificate per user. For these, make sure that the common name matches the users login.

```
$ openssl req -new -nodes -out filename.req.pem -keyout private/filename.key.pem
-days 3650 -config conf/caconfig.cnf
(for MCI) -> CN: CLIENT_IP_OR_NAME
(for ECI) -> CN: IRIS_SERVER_NAME
(for Browser) -> CN: LOGIN
$ openssl ca -out newcerts/filename.cert.pem -days 3650 -config conf/caconfig.cnf
-infiles filename.req.pem
```

6. **Encrypt certificates**

   To encrypt certificates for secure storage on the Safer Payment instances run the following command:

   ```
   openssl rsa -des3 -in private/<filename>.key.pem -out private/<filename>.enc.key.pem
   ```

   It is recommended to do this for both, client and server certificates.

7. **Create a certificate revocation list**

   ```
   vim certs/ca.crl
   vim crl.confg
   ```

   Content of crl.confg

   ```
   [ ca ]
   default_ca      = CA_default                  # the default ca section

   [ CA_default ]
   dir             = ./                          # where everything is kept
   database        = $dir/index.txt              # database index file.
   certificate     = $dir/certs/cacert.pem       # the CA certificate
   crl             = $dir/certs/ca.crl           # the current CRL
   private_key     = $dir/private/cakey.pem      # the private key
   default_crl_days = 183
   ```

   ```
   $ openssl ca -config conf/caconfig.cnf -gencrl -out crl/crl.pem
   ```

8. **Configure client-side certificates in web browsers**

   Convert pem certificate to p12:

   ```
   openssl pkcs12 -export -out newcerts/filename.cert.p12 -inkey private/filename.key.pem
   -in newcerts/filename.cert.pem -certfile certs/cacert.pem
   ```

   Next import the client-side certificates in your browser.

Configure cardholder data storage locations

This section describes the requirements for cardholder data storage locations and how to configure them.

PA-DSS requirement 9 mandates that cardholder data must not be stored on a server that is connected to the internet.

To comply with this requirement, you must do one of the following:

- Disable access from the internet to the server that hosts the Safer Payments instances. Remote VPN access (PA-DSS requirement 10) is not considered as access from the internet, if the VPN tunnel does not end directly on a server that hosts the Safer Payments instances.
- Place the data storage directories on a separate server computer, which is not connected to the internet. SAN storage is not accepted as a separate server computer by PCI DSS, unless network traffic between the SAN server and the server that hosts the Safer Payments instance is controlled by a firewall.

Note:

- You must disable the **locate** commands for the separate server computer. See "Disable locate for Safer Payments folders" on page 19 for details.
- Changes to the file storage locations are processed after a restart of a Safer Payments instance. Thus you can move the files while the instance is offline.

Configuration steps

Safer Payments can store cardholder data in a number of locations. To identify and adjust these locations, follow the steps.

1. Go to the user interface of Safer Payments
2. Click the **Administration** tab
3. Select **Cluster** from the left navigation pane.
4. Select a cluster instance from the table.
5. Scroll down to the **Local Storage** window.
6. For each cluster instance, the following directory locations can contain encrypted cardholder data:

Table 1. Default cardholder storage locations

Path name (default)	PAN stored as	PAN contained in
Case archive (arc)	encrpyted	archived cases
Configuration	encrpyted	conditions
Disk data cace (ddc)	encrpyted	attributes and indices
Email (eml)	masked, PANs are potentially also encrypted	notifications and case actions
FLI buffer (fli)	encrpyted	FLI messages
Case investigation (inv)	encrpyted	cases
Relational database interface (rdi)	masked	DML statements
User (usr)	encrpyted	user preferences
Log (log)	masked	log messages

7. You can now change the directory locations according to your configuration.

   Note: The locations are different for each Safer Payments instance. You must adjust the locations individually for each cluster instance.

Copy settings to the other Safer Payments instances

When you have configured the initial cluster configuration as described in "Configure cardholder data storage locations" on page 16, you must shut down the first Safer Payments instance.

1. Go to the **Cluster Settings** window.
2. Right-click the mouse and select **shutdown** from the menu.
3. Copy the files `cluster.iris` and `settings.iris` from the `cfg` subdirectory of the first Safer Payments instance to the `cfg` subdirectories of all the other Safer Payments instances.
4. Delete the entire contents of the `fli` directory of the instance that you have used to create the files.

5. You must also assign SSL certificate files to each Safer Payments instance, as described in “Configure SSL encryption” on page 10.
6. When you have copied the configuration files to the other Safer Payments instances, you can now also start these instances as described in “Start and stop Safer Payments instances” on page 35.

Note: Do not send any cardholder data to Safer Payments yet, as configuration according to PCI DSS requirements is not complete.

Configure for operational use

This sections describes how to configure Safer Payments for operational use.

Event logging

Users typically do not have access to the Safer Payments server. Safer Payments communicates its activities and status is generated through log messages.

In standard operations, these log messages are written to files where they can be viewed either directly using a text editor, by system tools, or Safer Payments itself.

Safer Payments contains a fully configurable event logging engine that supports three types of logging targets. The system and audit logs are Safer Payments logs. That is, Safer Payments has built-in viewer facilities to read these log messages.

System log
The system log informs about events relevant to technical operations of Safer Payments.

Audit log
The audit log traces relevant user activities.

External logs
External logs are sent to the operating system. In Linux/Unix operating systems such as RHEL, Safer Payments feeds external log messages to the local `syslogd` as `"IRIS_`*n*. *n* is the ID of the Safer Payments instance, as defined by the command line parameter. External logging is mandatory in PCI DSS compliant environments to facilitate centralized logging, and must thus be activated.

Make sure that all PCI DSS relevant log messages are forwarded to centralized logging as described in “Change log message settings” on page 37.

Note: If you use an IBM MQ server to deliver data to Safer Payments, you must ensure that all relevant log messages are forwarded as well.

Swap disk configuration

Safer Payments is designed to not use swap memory in running mode. However, the operating system can use swap memory for some operations.

In this case, PAN data, which is decrypted in RAM might temporarily be written to the swap file on disk. You must wipe all swap data securely after each new system restart or use an encrypted swap disk. You must also disable indexing of file contents.

Wipe swap disk script

Note: Use this approach only, if swap disk encryption is not possible for certain reasons.

1. To find out the right path of your swap disk partition enter:
   ```
   # fdisk -l#
   cat /proc/swaps
   ```
2. If you have your swap partition name, write a small script, which runs on every startup using `sswap`, where */dev/sdaX* must be replaced by the path shown in the previous step.
   ```
   # swapoff /dev/sdaX
   # sswap -vll /dev/sdaX
   # swapon /dev/sdaX
   ```
3. Add this code to a script. For example, to: `/usr/local/sbin/wipeSwap.sh`
   ```
   chmod +x /usr/local/sbin/wipeSwap.sh
   ```
4. Add the script name `/usr/local/sbin/wipeSwap.sh` at the bottom of your init script `/etc/rc.local`.

Encrypt swap disk

With this preferred approach, you do not have to wipe out your swap on each system start.

1. Edit `/etc/fstab` to reflect the changes. Comment or delete previous swap entries before you add the new entry.
   ```
   # vim /etc/fstab
   /dev/mapper/swap none swap defaults 0 0
   ```
2. Create a `/etc/crypttab` file, and add the swap parameters.
   ```
   # vim /etc/crypttab
   swap /dev/volume /dev/urandom swap,cipher=aes-cbc-essiv:sha256
   ```

   Depending on your volume, group names, and layout, change the path to suit your needs. In most cases you only have to replace *volume* with the path you have commented or deleted in step 1. The encryption system then uses AES and SHA256 bit encryption during startup, with a random key. A new key is generated each time that the server is started.
3. Reboot the server to enable swap disk encryption.
4. Verify that swap disk encryption is enabled with the **lsblk** command.
   ```
   # lsblk
   ```

Disable locate for Safer Payments folders

locate is a daemon, which creates a database with file contents. Make sure that either applies:

- **locate** is not installed.
- **locate** is disabled on the operating system.
- All Safer Payments folders are excluded from the **locate** search paths.

To change the **locate** search paths, edit the `/etc/updatedb.conf` file and add the Safer Payments folder to **PRUNEPATHS**.

Assuming `PRUNEPATHS` is set to:

```
PRUNEPATHS="/afs /media /net /sfs /tmp /udev /var/cache/ccache
/var/spool /var/tmp /mnt/iris"
```

And you have installed Safer Payments to /myInstallation as described in "Run the installer to extract the installation image" on page 3.

Change PRUNEPATHS to:

```
PRUNEPATHS="/afs /media /net /sfs /tmp /udev /var/cache/ccache
/var/spool /var/tmp /myInstallation"
```

Note:

- If you have adapted Safer Payments file storage locations during cluster configuration as described in "Configure cardholder data storage locations" on page 16, you must also add folders that are used outside /myInstallation to PRUNEPATHS. Keep in mind that file locations can be configured differently per cluster instance.
- If any such folders are located on separate servers, PRUNEPATHS must be adjusted on those servers as well.

Miscellaneous system settings

The following extra configuration steps are necessary before you start Safer Payments.

1. Log in as root.
2. The default number of maximum open file descriptors on CentOS/RHEL systems is 1024 per process for normal users. To verify the limit that is valid for your system, run

   ```
   # cat /proc/sys/fs/file-max
   ```
3. Open the file /etc/security/limits.conf
4. If the recommended limit of 16384/32768 is valid for your system, add the lines

   ```
   SPUser hard nofile 32768
   SPUser soft nofile 16384
   ```

   Where *SPUser* is the name of the user account that you intend to run Safer Payments under.
5. To enable Safer Payments to run priority-based thread scheduling, you must also add

   ```
   SPUser - rtprio 20
   ```
6. Safer Payments starts numerous CPU threads for parallel processing of messages and simulations. To ensure, that the operating system can handle all threads, you must increase the number of maximum user processes. To do so also add the line

   ```
   SPUser - nproc 8192
   ```
7. Summary of necessary changes to /etc/security/limits.conf. In this example, the user name of the process running Safer Payments is *SPUser*.

   ```
   SPUser hard nofile 32768
   SPUser soft nofile 16384
   SPUser - rtprio 20
   SPUser - nproc 8192
   ```
8. Save /etc/security/limits.conf and reboot.

Firewall settings

Before you start Safer Payments, check your firewall settings to allow IP messaging between the Safer Payments cluster instances and other systems.

To change your local firewall settings, use **system-config-firewall-tui** on RHEL 6 or **firewall-cmd** on RHEL7.

For general information on how to secure your operating system see:

- RHEL 6

  https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/index.html
- RHEL 7

  https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/index.html
- Oracle Linux 7

  https://docs.oracle.com/cd/E52668_01/E54670/E54670.pdf

Deferred writing and ultra-large memory configuration

Transparent Huge Pages (THP) is a Linux memory management system that reduces the overhead of Translation Lookaside Buffer (TLB) lookups on machines with large amounts of memory by using larger memory pages.

If you are using ultra-large main memory configurations and deferred writing (Safer Payments UI / Administration / System Configuration / Deferred Writing) on Linux operating systems, you might experience faster restarts and more stable latencies when disabling transparent huge pages. Transparent huge pages might block the memory for seconds, when it is defragmenting the RAM. In this time, it is not possible to make even small memory allocations.

Be careful with this setting. It can also result in slower overall message computation, when deferred writing is disabled.

To disable transparent huge pages temporarily, run the following command:

```
echo never > /sys/kernel/mm/transparent_hugepage/defrag
echo never > /sys/kernel/mm/transparent_hugepage/enabled
```

Next restart Safer Payments. To check, if transparent huge pages are disabled run the following command:

```
cat /sys/kernel/mm/transparent_hugepage/enabled
```

The output is:

```
always madvise [never]
```

To disable transparent huge pages permanently in RHEL 6, add both lines to the Safer Payments start script directly after `echo -n "starting $PROG: "` as follows:

```
echo -n "starting $PROG: "
echo never > /sys/kernel/mm/transparent_hugepage/defrag
echo never > /sys/kernel/mm/transparent_hugepage/enabled
```

In RHEL 7, you can disable transparent huge pages with the command **tuned**.

See https://access.redhat.com/solutions/1320153 for details.

You can query the current active profile with:

```
# tuned-adm active
Current active profile: latency-performance
```

To create a customized profile, create a new directory in the /etc/tuned directory with the wanted profile name.

```
# mkdir /etc/tuned/myprofile-nothp
```

Next, create a new tuned.conf file for *myprofile*-nothp, and insert the new tuning information.

```
# cat /etc/tuned/myprofile-nothp/tuned.conf
[main]
include= latency-performance
[vm]
transparent_hugepages=never
```

Next, make the script executable:

```
# chmod +x /etc/tuned/myprofile-nothp/tuned.conf
```

Next, enable *myprofile*:

```
# tuned-adm profile myprofile-nothp
```

This change immediately takes effect and persists a reboot.

Increase virtual memory map size

Linux restricts the maximum number of memory maps per process. The default is 65535 on RHEL 6/7.

This value might be enough for most Linux applications, but depending on the Safer Payments configuration and internal data allocation sizes, Safer Payments might actually need more memory maps. If the application has reached its maximum number of memory maps, a "cannot allocate memory" error message and other errors occur, even if enough free RAM is available in the system. To avoid this situation, run:

```
echo 1048576 > /proc/sys/vm/max_map_count
```

This command temporarily applies the new maximum number of virtual memory maps to 1048576. If you want to increase the value permanently, you must add this value as vm.max_map_count=1048576 to the file /etc/sysctl.conf after server restart. To check if the configuration was applied correctly run:

```
cat /proc/sys/vmmax_map_count
```

If you have a very large configuration, you should monitor the current number of memory maps during high load. If the number of memory maps exceeds half of its maximum value, increase this value. To monitor the number of memory maps for your Safer Payments process, run:

```
cat /proc/IRIS_PID/maps | wc -l
```

Decrease swappiness

Safer Payments produces larger latencies, if the system uses swap memory.

Therefore, it is recommended to reduce the swappiness on a Linux system.

- To temporarily change the setting run

  ```
  sysctl -w vm.swappiness=1
  ```

- To permanently change the setting add

  ```
  vm.swappiness=1
  ```

  to /etc/sysctl.conf

RHEL service script

It is possible to start Safer Payments as a service on Linux.

For RHEL 7, the start script is located in `/etc/systemd/system/iris.service`.

```
[Unit]
Description=Safer Payments Service
After=network.target
[Service]
Type=simple
User=SPUser
ExecStart=/usr/bin/iris rootpath=/opt/ibm/safer_payments/myInstallation id=1
Restart=no
TimeoutSec=0
TimeoutStartSec=0
TimeoutStopSec=0
SendSIGKILL=no
KillSignal=SIGTERM
LimitNOFILE=32768
LimitNPROC=8192
LimitRTPRIO=20
[Install]
WantedBy=multi-user.target
```

- To enable Safer Payments to run at startup, enter: `systemctl enable iris.service`
- To start Safer Payments, enter: `systemctl start iris.service`
- To stop Safer Payments, enter: `systemctl stop iris.service`
- To query if Safer Payments is running, enter: `systemctl status iris.service`

For RHEL 6, the start script is located in `/etc/init.d/irisd`.

```
# Startup script for Safer Payments server
#!/bin/bash
# Startup script for IRIS Server
#
# processname: irisd
#
### BEGIN INIT INFO
# Provides:            irisd
# Short-Description:   Start and stop IRIS Server
# Description:         This script starts and stops the IRIS Server.
#                      It is necessary that only one instance is running on this machine.
#                      Modify this script to run multiple machines on one instance.
### END INIT INFO

# Source function library.
. /etc/rc.d/init.d/functions #centos
# . /lib/lsb/init-functions #debian

IRISD=/usr/bin/iris
DAEMON_OPTS="rootpath=/mnt/iris id=1"
PROG=IRIS
USERNAME="SPUser"
NOHUP_OUTPUT=/mnt/iris/nohup.out # Console output redirect
START_WAIT=5          # Wait x seconds to ensure that Safer Payments has loaded some configurations.
RETVAL=0

iris_is_running() {
 pidof -sx $IRISD > /dev/null
 return $?;
}

start() {
 if iris_is_running; then
  echo "Server already started"
  exit 1
 fi
 echo -n "Starting $PROG: "
 nohup su $USERNAME -c "$IRISD $DAEMON_OPTS "&> $NOHUP_OUTPUT&
 sleep $START_WAIT
 if iris_is_running; then
```

```
  success $PROG
  echo
  return 0;
 else
  failure $PROG
  echo
  return 1;
 fi
}

wait_iris_stop() {
 if [ $RETVAL = 0 ]; then
  while iris_is_running
   do
   sleep 1
  done;
  success $PROG
 else
  failure $PROG
 fi
}

stop() {
 echo -n $"Stopping $PROG "
 RETVAL=0
 if iris_is_running; then
  echo -n $" (sending SIGTERM): "
  kill -TERM $(pidof -sx $IRISD)
  RETVAL=$?
  wait_iris_stop
 else
  echo -n $" (is not running): "
 fi
 echo
 return $RETVAL
}


kill_instance() {
 echo -n $"Killing $PROG "
 RETVAL=0
 if iris_is_running; then
  echo -n $" (sending SIGTERM): "
  kill -KILL $(pidof -sx $IRISD)
  RETVAL=$?
  wait_iris_stop
 else
  echo -n $" (is not running): "
 fi
 echo
 return $RETVAL
}
# See how we were called.
case "$1" in
 start)
  start
  ;;
 stop)
  stop
 ;;
 kill)
  kill_instance
 ;;
 version)
  $IRISD release
  RETVAL=$?
 ;;
 status)
  status $IRISD #centos
  #status_of_proc $IRISD #debian
  RETVAL=$?
 ;;
 restart)
  stop
  start
 ;;
 condrestart|try-restart)
 if iris_is_running; then
  stop
  start
```

```
 fi
 ;;
 help)
  echo $"Usage: $PROG {start|stop|kill|version|restart|condrestart|try-restart|status|help}"
  RETVAL=2
 ;;
esac

exit $RETVAL
```

Data encryption

This section describes how to enable data encryption, to be compliant with PA-DSS.

Activate data encryption - step 1

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **System configuration** from the left navigation pane.
3. Scroll down the **System Configuration** window to locate the window titled **Encryption**.

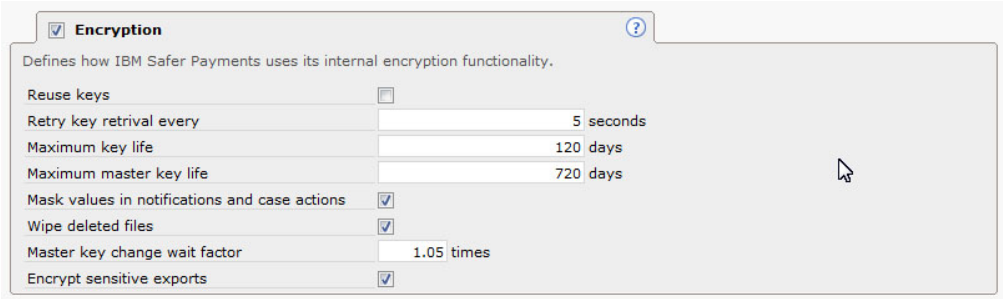


Figure 7. Encryption window

4. Clear the **Reuse keys** check box.
5. Select the **Mask values in notifications**, **Wipe deleted files**, and **Encrypt sensitive exports** check boxes.

Encryption covers the actual production data and certain parts of the configuration where PANs are expected, for example, in conditions and audit trails. Other parts of the configuration are not encrypted. This implies that you must never store clear PAN in any name or comment field of Safer Payments.

The PA-DSS standard recommends defining a maximum cryptoperiod after which a key must be replaced with a new one. See “Enforce regular key changes” on page 52 for details.

According to PA-DSS requirement 2.3, PANs must be rendered unreadable anywhere they are stored. Therefore, you must enable **Encrypt sensitive exports**.

Using a secure wipe tool

Various PCI DSS requirements demand the use of a secure wipe tool to securely delete sensitive authentication data and cardholder data.

According to PA-DSS requirement 1.1.4, the disk wipe tool must be in accordance with industry accepted standards for secure deletion. The National Security Agency, for example, maintains a list of approved products.

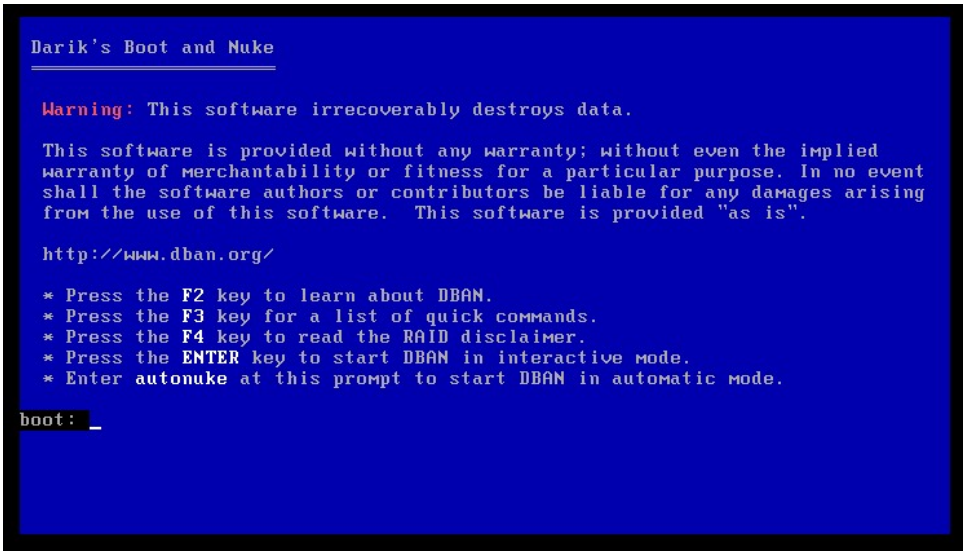
To securely wipe entire hard disks, you can use the “DBAN” tool.

To securely delete single files or directories you can use the Linux tool “Wipe”.

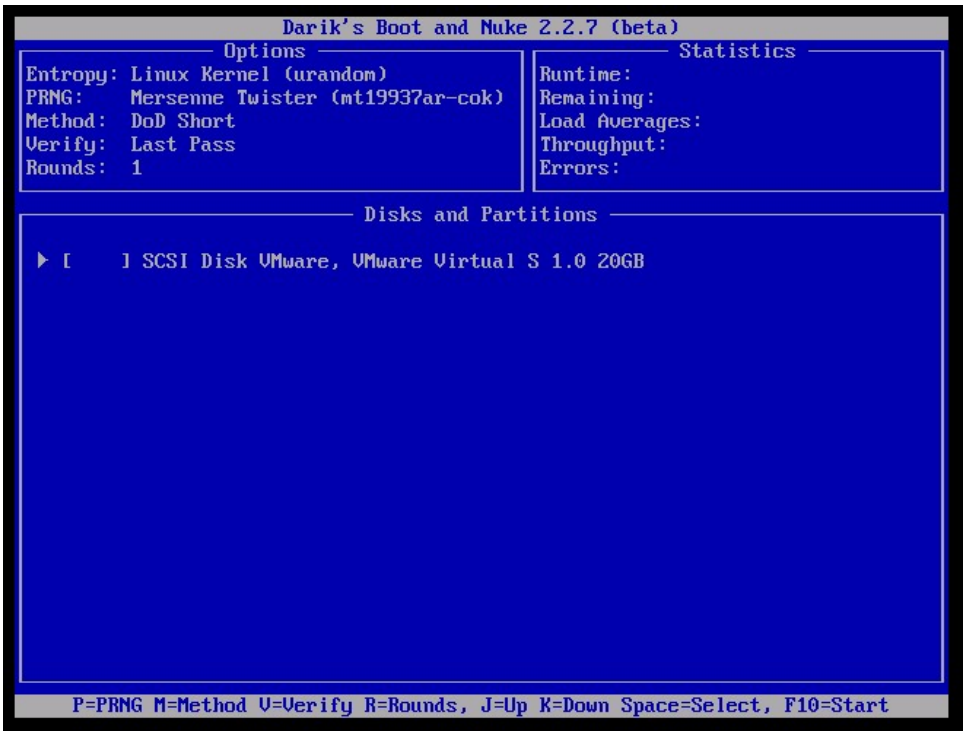
Using the DBAN tool

You can download DBAN from http://www.dban.org/.

1. Create a CD with the ISO image of DBAN.
2. Boot the computer that hosts the device you want to wipe securely.
3. Press the **ENTER** key to start DBAN in interactive mode.

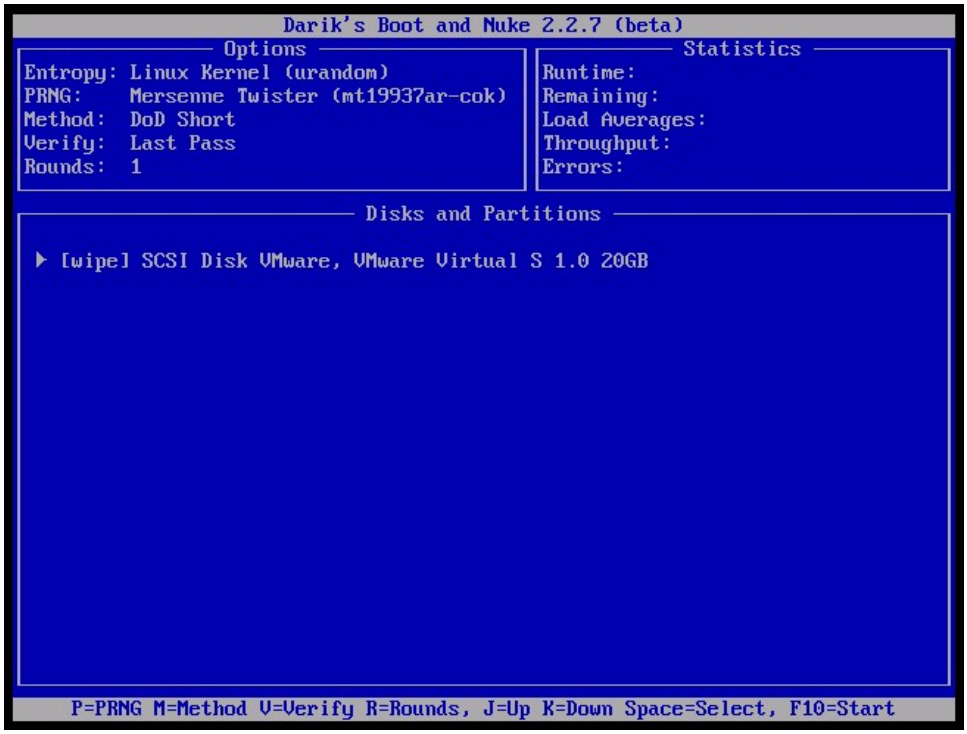


4. Type M and select the **DoD Short** method.

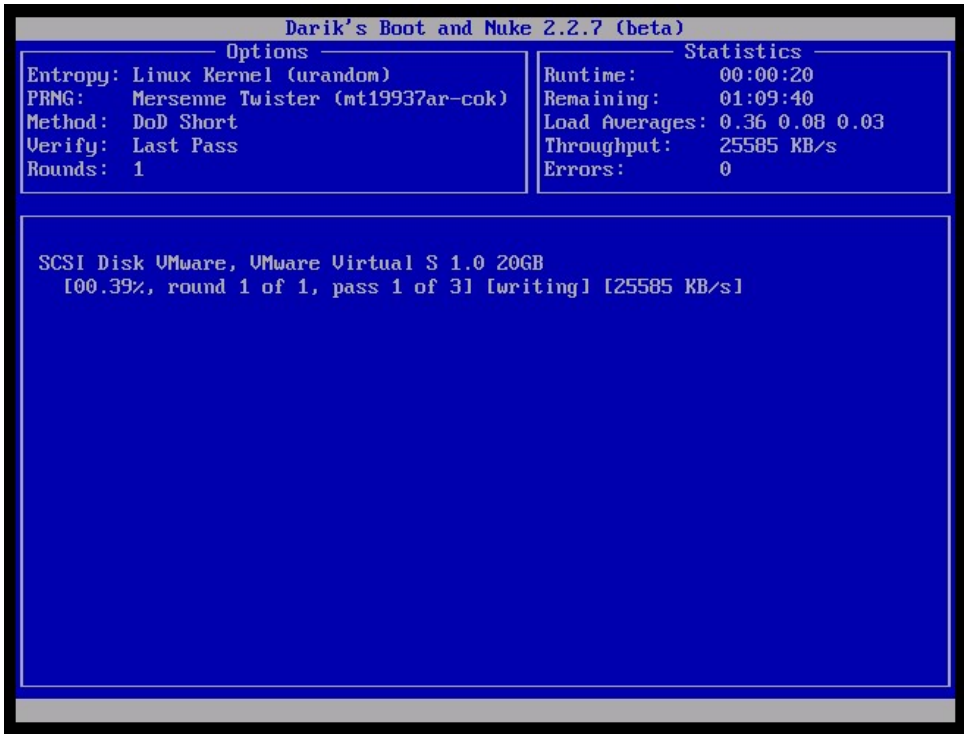


5. Select the disk or partition you want to wipe by using the up (J) and down (K) keys to move to the entry.
6. To confirm your selection, press the space bar.

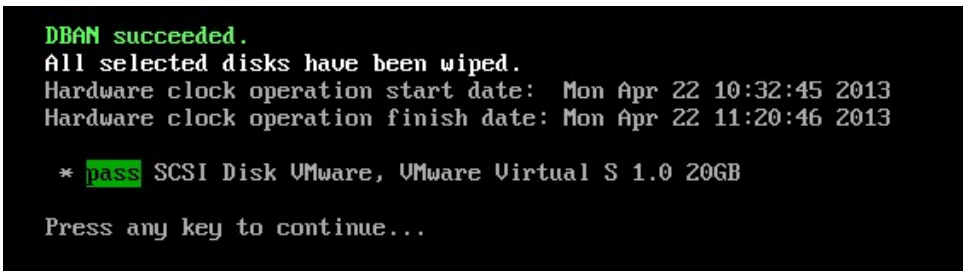
7. To start wiping the disk, press F10.



8. The disk is now being wiped.



9. Make sure a dialog is displayed that confirms a successful wipe.



Using the Wipe tool

You can download Wipe from http://wipe.sourceforge.net/.

You can use the Wipe tool to securely delete single files or directories.

For example, to securely delete file *myfile.txt* run:

```
wipe -Sr -p3 myfile.txt
```

Decommission a Safer Payments instance or cluster

If a Safer Payments cluster instance or an entire Safer Payments cluster is decommissioned, you must securely delete all cardholder data by using a disk wipe tool.

Safer Payments stores cardholder data in several locations. These locations are identified and configured as described in "Configure cardholder data storage locations" on page 16.

Archived data and backups that are created by third-party applications are not in reach of the Safer Payments software itself. Therefore, they are not securely deleted automatically by Safer Payments. This aspect of this requirement must be met by organizational procedures.

Set up key entry and key management users

Before you proceed, you must set up user accounts in Safer Payments, which have sufficient privileges to manage data encryption.

According to PA-DSS requirement 2.5.6, two key holders are required. One must be granted the global privilege **left public key entry**, the other one the global privilege **right public key entry**.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **User accounts** from the left navigation pane.
3. Click the  icon to create a new user account.
4. The **New User Account** window is displayed.

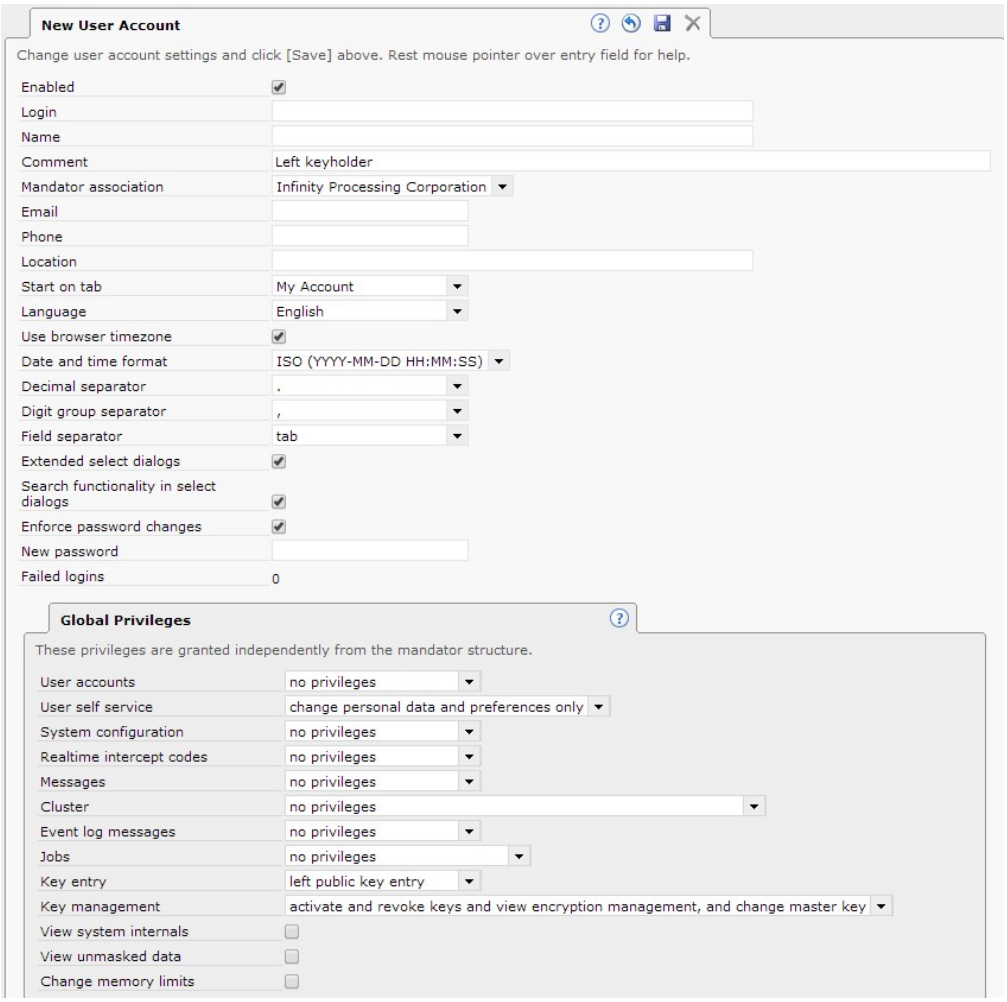


Figure 8. New User Account window

5. On the **Global Privileges** window, select **left public key entry** in the **Key entry** field.
6. Repeat step 3 for the second user account and select the **right public key entry** in the **Key entry** field.

   Note: You cannot assign both left and right public key entry privileges to a single user.
7. There must be a user that is granted the privilege to **activate keys** in the **Key management** field. This privilege can be granted to one or both key holders, or any other user.
8. In the **Key management** field, assign the **change masterkey** privilege to a user. This is required for changing the master key as described in "Change the master key" on page 54.

   Important: The person who sets up the user accounts must make sure that the check box **enforce password changes** is selected.

Activate data encryption - step 2

Before you can use encryption in Safer Payments you must generate keys.

Key generation and distribution is described in "Key generation" on page 46.

After you have generated and distributed keys, you must

- Either restart all Safer Payments instances.
- Or
  1. Click the **Administration** tab.
  2. Select **Encryption keys** from the left navigation pane.
  3. Click the  icon to reload private keys from disk.

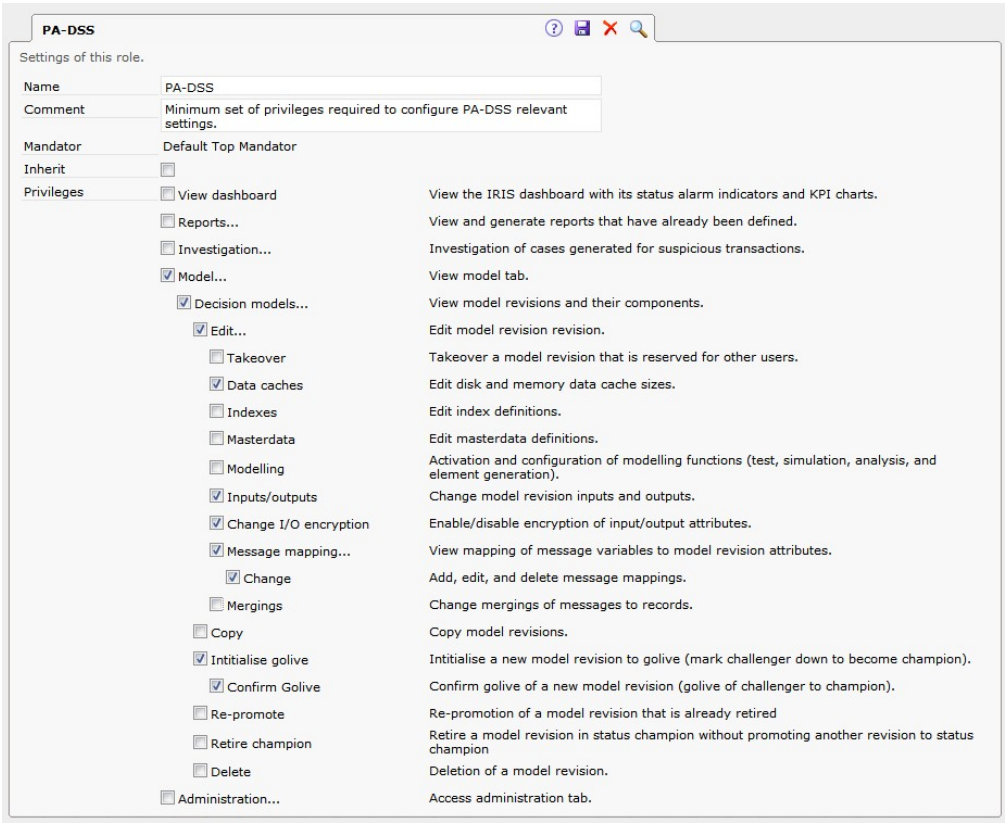
Enable cardholder data encryption

In the next step, you must activate PA-DSS compliant encryption of cardholder data to be stored in Safer Payments. To comply with PA-DSS requirement 1 do not process sensitive authentication data.

If you intend to store the Primary Account Number (PAN) in Safer Payments, you must enable encryption for this data attribute in Safer Payments as defined in PA-DSS requirement 2.3.

Attribute names can be chosen freely in Safer Payments. However, in this documentation the attribute for the Primary Account Number is named “PAN”.

1. Log on to Safer Payments with a user account as described in “Start the first Safer Payments instance” on page 8 that has been granted at least the following privileges:



Note: Refer to the online documentation for details of user access right administration.

Figure 9. PA-DSS settings window

2. On the Safer Payments user interface, click the **Model** tab.

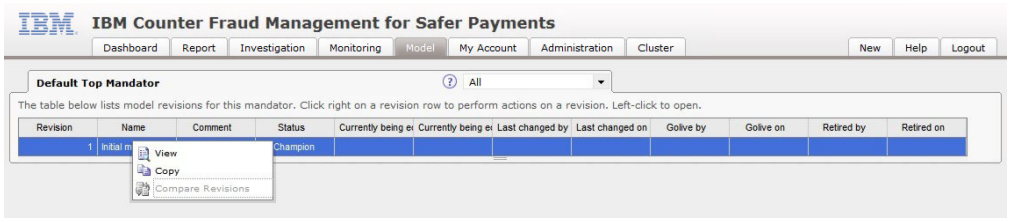


Figure 10. Model - Default Top Mandator window

3. The Default Top Mandator is displayed.
4. Right mouse-click on the **Champion** entry.
5. On the pop-up selection box, select **Copy**.
6. Click the newly created **Challenger** entry.
7. Select **Inputs** from the left navigation pane.

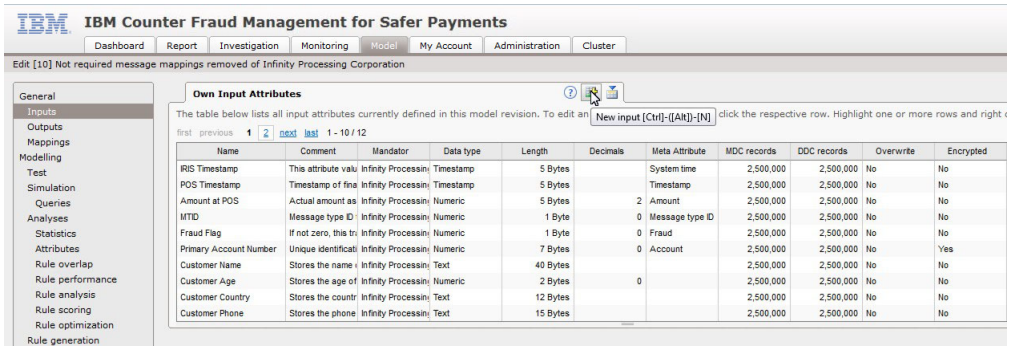


Figure 11. Create new attribute window

8. Click the  icon to create a new attribute.
9. The **New Attribute** window opens.

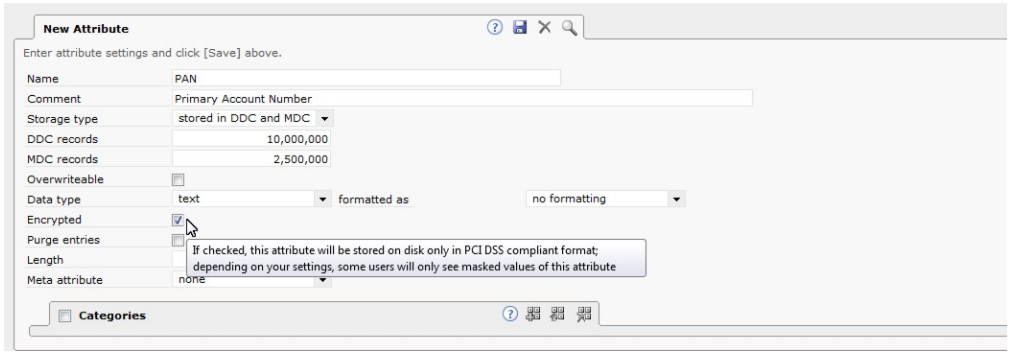


Figure 12. New attribute settings window

Note: To be PA-DSS compliant you must now enable encryption for the PAN attribute. For all other sections not relevant for PA-DSS refer to the online documentation.

10. Enter **PAN** in the Name field and select the check box in the **Encrypted** field.

11. Save the new attribute by clicking the  icon.
12. You can now define other attributes that are required for your specific Safer Payments application. To comply with PA-DSS make sure that no sensitive authentication data is defined.
13. Next select **General** from the left navigation pane.

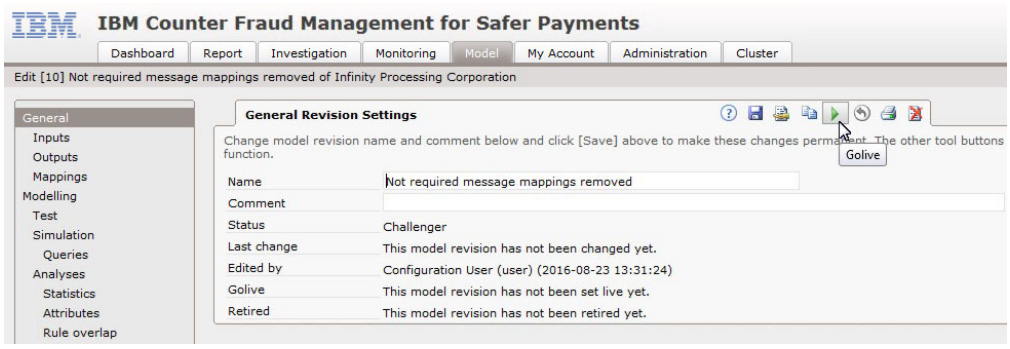


Figure 13. Activate changed revision

14. Click the  icon. The decision model is now being activated and all data that is stored in the PAN attribute is encrypted.

Key life expiration

PA-DSS requirement 2.5.4 mandates regular key changes. Therefore, Safer Payments automatically shuts down, if the maximum key life of a key is reached. You can implement Status Alarm Indicators, that alert you of keys that are expiring soon. See "Enforce regular key changes" on page 52 for details on key changes and setting up SAIs.

Miscellaneous Safer Payments configuration settings

To meet various PA-DSS requirements, further configuration settings must be changed.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **System configuration** from the left navigation pane.
3. Scroll down the **System Configuration** window to locate the **User Account** window.

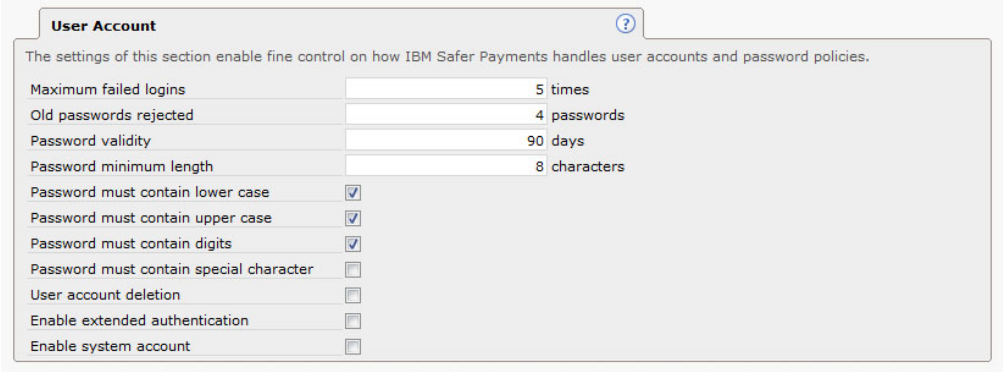


Figure 14. User account settings

4. Select the **Password must contain lower case**, **Password must contain upper case**, and **Password must contain digits** check boxes.
5. Scroll further down to locate the **Application Programming Interface** window.

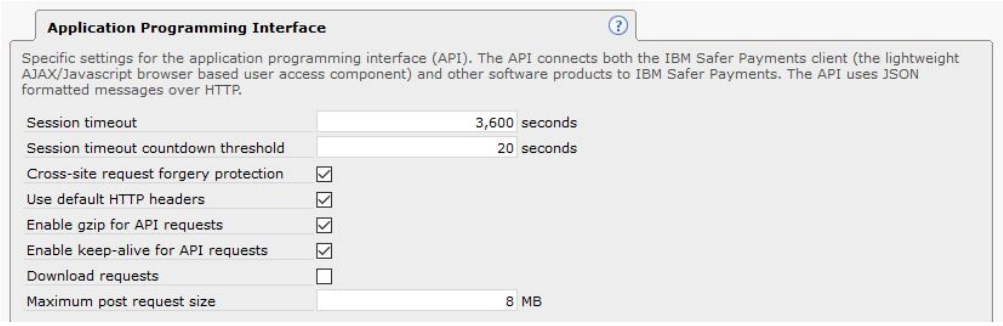


Figure 15. API settings

6. Select the **Cross-site request forgery protection** check box.

   Note: If you want to use tested default and secure HTTP headers, select the **Use default HTTP headers** check box.
7. Scroll further down to locate the **Message Tracing** window.

Figure 16. Message Tracing settings

8. Clear the **Show IP data** check box.

   Note: Show IP data needs to be disabled to comply with PA-DSS requirement 2.3.
9. Scroll further down to locate the **Miscellaneous** window.
10. Verify that the SSL cipher list has the following entries:

    ```
    ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-
    ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-
    AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-
    ```

```
SHA256:DHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:
ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES256-
SHA:ECDHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:
DHE-RSA-AES256-SHA256:DHE-RSA-AES256-SHA:ECDHE-ECDSA-DES-CBC3-SHA:
ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES128-GCM-SHA256:
AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-SHA:AES256-SHA:
DES-CBC3-SHA:!DSS:!aNULL:!eNULL:!EXPORT:!DES:!3DES:!RC4:!MD5:!PSK:
!aECDH:!EDH-DSS-DES-CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-DES-CBC3-SHA
```

Note: This list might be outdated because new security leaks have been discovered in the meantime. The OpenSSL website provides regular security advisories, including information about potential security leaks.

http://www.openssl.org/

11. If you are using the relational database interface, you must enable masking values of encrypted attributes.
12. Go to the Mandator window, scroll down to the **Relational Database Interface** window, and open it by selecting the check box.

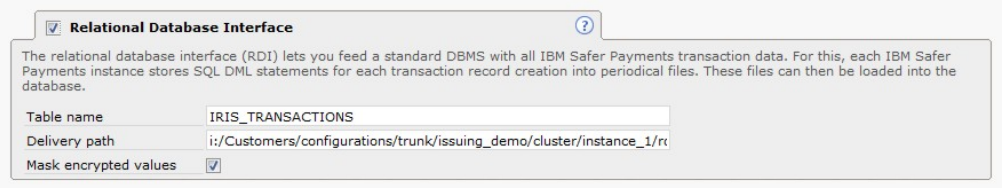


Figure 17. Relational Database Interface settings

13. Select the **Mask encrypted values** check box.

    Note: You must do this for all mandators.

Using NFS for BDI job files

Note: This topic applies only if you are using RHEL 7.

If you are using NFS to share batch files between multiple instances, you must change the mount parameters to avoid blocking locks in Safer Payments. If you don't change the parameters and use RHEL 7, you might have blocking jobs and hanging golives on instances that were running jobs previously.

Safer Payments tries to lock job files to prevent modification during job run. It uses a non-blocking call to perform this lock. However, newer NFS clients might still block during this non-blocking call, which can result in a malfunction of production systems.

To prevent this you must add `nolock` as an additional parameter in the file system table (fstab). For example:

```
"rw,bg,vers=3,tcp,timeo=600,rsize=65536,wsize=65536,nolock"
```

You can change all other parameters according to your needs, but make sure that you are using `nolock` when using NFS for BDI files.

Operation of Safer Payments

This section describes various tasks that need to be run during regular operation of Safer Payments.

Start and stop Safer Payments instances

This topic describes how to start and stop Safer Payments during regular operation.

Start a Safer Payments instance

Open a console window on the server and run:

```
iris console id=i
```

from */myInstallation*/cfg.

- The `console` parameter activates event log message output on the console window.
- *myInstallation* is the home directory.
- The parameter *i* is the ID of the instance that you want to start.

  Note: Both home directory and instance ID have been defined in "Start the first Safer Payments instance" on page 8.

Safer Payments now starts up. You must enter the SSL certificate password when you are prompted.

Each Safer Payments instance now attempts to fetch the password for the encryption keys from its sister instances. If no other instances are running yet, the encryption keys must also be entered before full operation can start. See "Activate keys" on page 50 for details on key entry and activation.

Stop a Safer Payments instance

To stop Safer Payments, you can use a SIGTERM command. Safer Payments catches SIGTERM signals and performs a clean shutdown, similar to the API shutdown command.

Open a console window on the server and run:

```
killall iris
```

To immediately stop a Safer Payments process, you can use the SIGKILL signal. Only use the SIGKILL signal, if Safer Payments does not properly shut down after a SIGTERM command:

Open a console window on the server and run:

```
killall -9 iris
```

Securely delete outdated index entries

For certain functions, Safer Payments requires indexing certain data attributes.

If you require an index on the PAN attribute, old index attributes must be securely deleted after the retention period defined in "Preliminary considerations" on page 1.

Safer Payments can be configured to securely delete outdated index entries automatically, by enabling the outdated entries setting.

1. On the safer Payments user interface, click the **Model** tab.
2. Right mouse-click the **Champion** entry.
3. On the pop-up selection box, select **Copy** .
4. Click the newly created **Challenger** entry.
5. Click **Indexes** on the left navigation pane.
6. On the **Own Indexes** window, click the **PAN** entry.

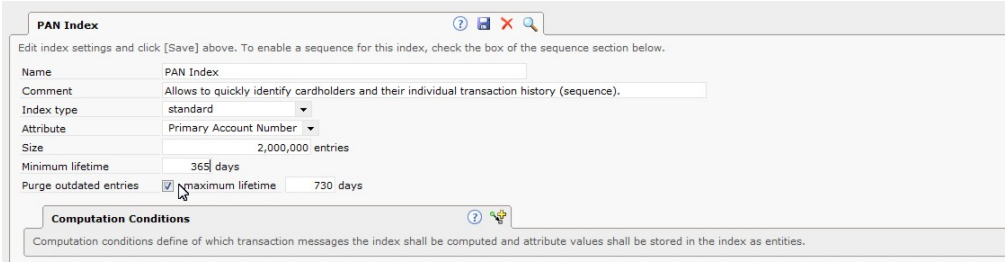


Figure 18. PAN Index window

7. Select the **Purge outdated entries** check box and enter a maximum lifetime.

Case archiving

Cardholder data must be securely deleted after the retention period defined in "Preliminary considerations" on page 1. If you enable case investigation, you must configure cases to be archived in Safer Payments no later than the end of the retention period.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **System configuration** from the left navigation pane.
3. Scroll down to locate the window titled **Case Investigation**.

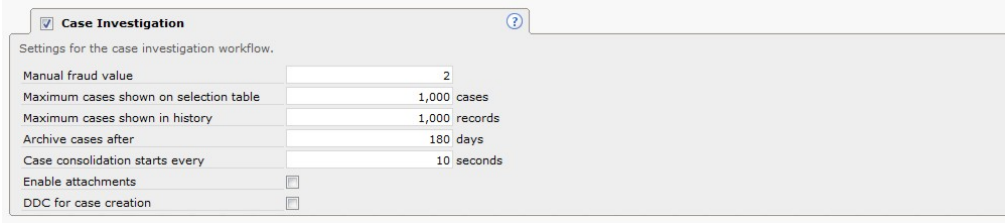


Figure 19. Case investigation settings window

4. In the **Archive cases after** field, you must enter a value that is equal or smaller to the retention period that you defined.

According to PA-DSS requirement 2.3, PANs must be rendered unreadable anywhere they are stored. Case attachments are stored decrypted. You must implement proper operational procedures to ensure that no PANs are stored in a case attachment or you must disable case attachments.

Change log message settings

To change the log message settings, you must perform the following steps:

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **Event log messages** from the left navigation pane.

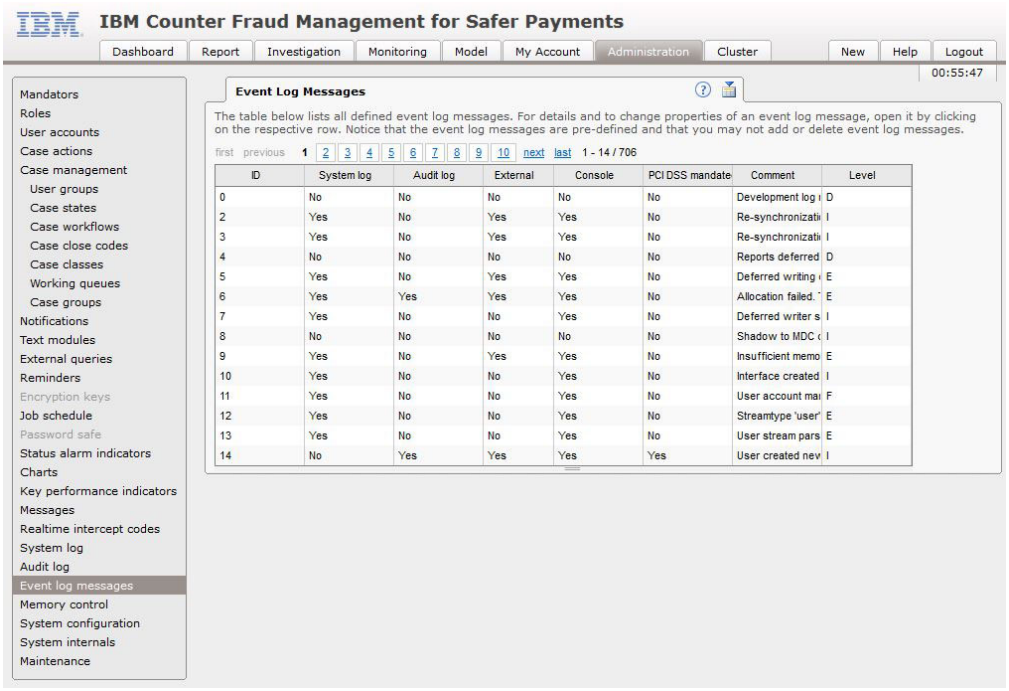


Figure 20. Event Log Message settings window

3. Click a log row, for example 100, to adjust the settings. This must be done for each log message individually.

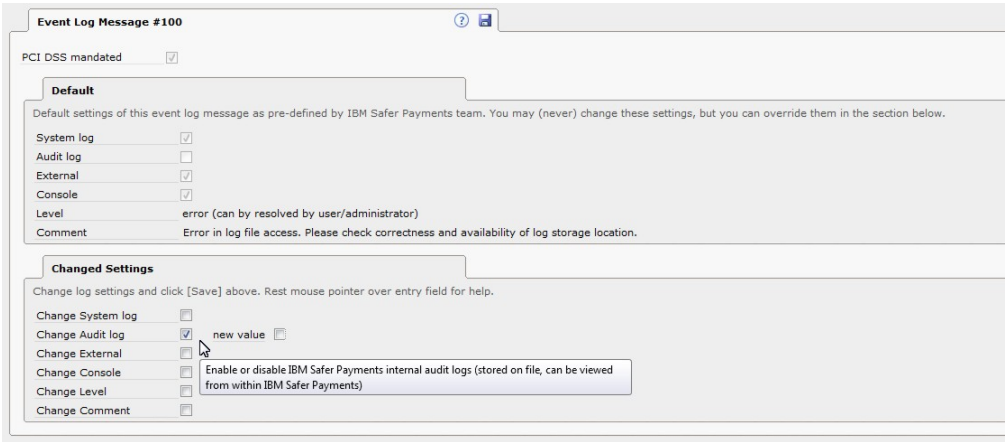


Figure 21. Case investigation settings window

4. Select the **Change** *xxx* box, for the value you want to change. The **new value** check box is displayed.
5. You can change the value by selecting or clearing the **new value** check box.

Note: Your central log server must collect all relevant log messages from the system log. You must implement an operational process within your organization to collect the relevant logs from the operating systems logs.

For PCI DSS compliance, a minimum set of log messages must be forwarded to centralized logging. To do so the **External** check box must be selected for each message. The following log messages are mandatory for PCI DSS compliance:

14, 15, 16, 17, 38, 41, 70, 71, 90, 91, 92, 93, 100, 129, 131, 157, 194, 195, 196, 197, 198, 199, 200, 201, 209, 210, 211, 212, 213, 229, 251, 322, 324, 325, 364, 365, 366, 367, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 417, 418, 419, 420, 425, 426, 427, 428, 429, 430, 434, 435, 436, 437, 448, 449, 456, 457, 460, 461, 462, 463, 468, 471, 490, 508, 509, 510, 511, 517, 518, 519, 520, 523, 524, 530, 531, 535, 536, 537, 541, 542, 543, 565, 573, 575, 577, 578, 580, 581, 585, 586, 587, 591, 594, 595, 598, 599

These log messages are set to the correct values by default after the initial installation of Safer Payments. To enable additional log messages for centralized logging you must select the **External** check box for each corresponding message.

Note: If you disable these log messages your are not PCI DSS compliant.

Archiving and backup

Safer Payments does not automatically remove archived cases, log messages, and so on, from the file system.

This avoids the loss of the data before it is being archived for auditing reasons, according to the licensee's requirements.

However, PCI DSS requires purging cardholder data after the customer-defined retention period. Therefore, the licensee must implement a backup process for those files, and ensure that the archived files are purged before the end of the retention period.

"Configure cardholder data storage locations" on page 16 lists the directory locations, which might contain encrypted cardholder data, and to which such a backup/purging process applies.

Note: Backups must also be handled according to PCI DSS requirements.

Set user privileges

According to PCI DSS, the PAN must be displayed masked only, unless there is a legitimate business need to see the full PAN.

Full PAN visibility is controlled by the **View unmasked data** global privilege for each user.

1. On the Safer Payments user interface, click the **My account** tab.
2. Scroll down to the **Global Privileges** window.

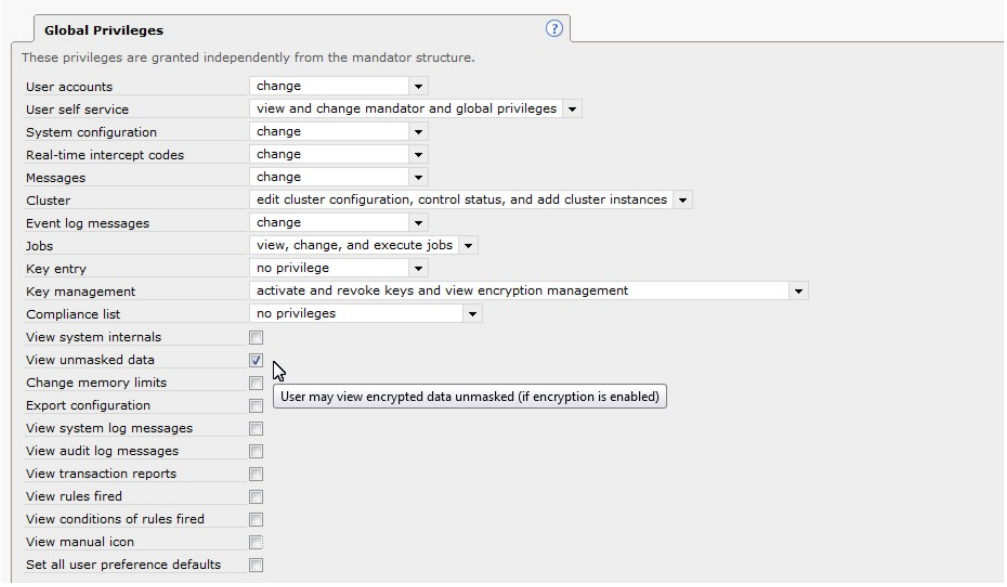


Figure 22. Global privileges window

3. Select **View unmasked data** check box.

In addition to global privileges, certain functions of Safer Payments can be accessed only by users with a legitimate business need. You can grant certain privileges to such users. More precisely, model revisions, report, and query definitions must be viewed only by privileged users. However, non-privileged users can still run reports and queries.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **Roles** from the left navigation pane.
3. Click the user role that you want to change in the roles table.

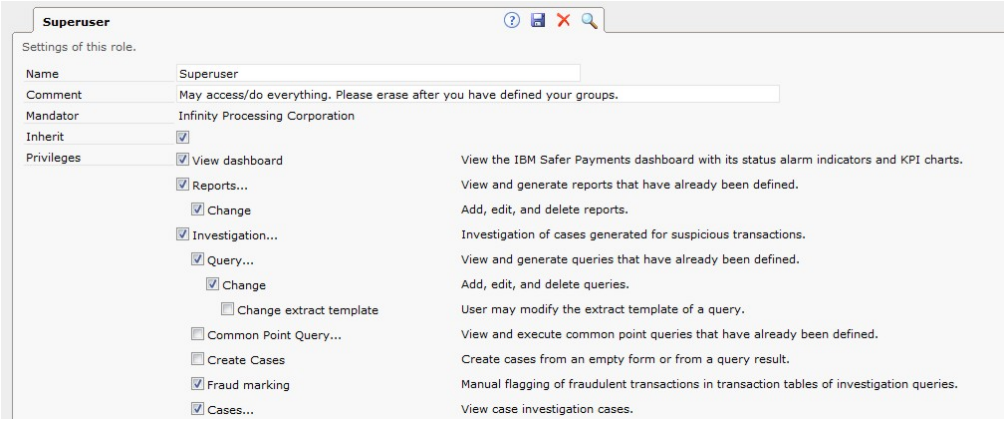


Figure 23. Superuser settings window

4. Change the privileges according to your requirements by selecting the appropriate check boxes.

Using a secure wipe tool

Various PCI DSS requirements demand the use of a secure wipe tool to securely delete sensitive authentication data and cardholder data.

According to PA-DSS requirement 1.1.4, the disk wipe tool must be in accordance with industry accepted standards for secure deletion. The National Security Agency, for example, maintains a list of approved products.

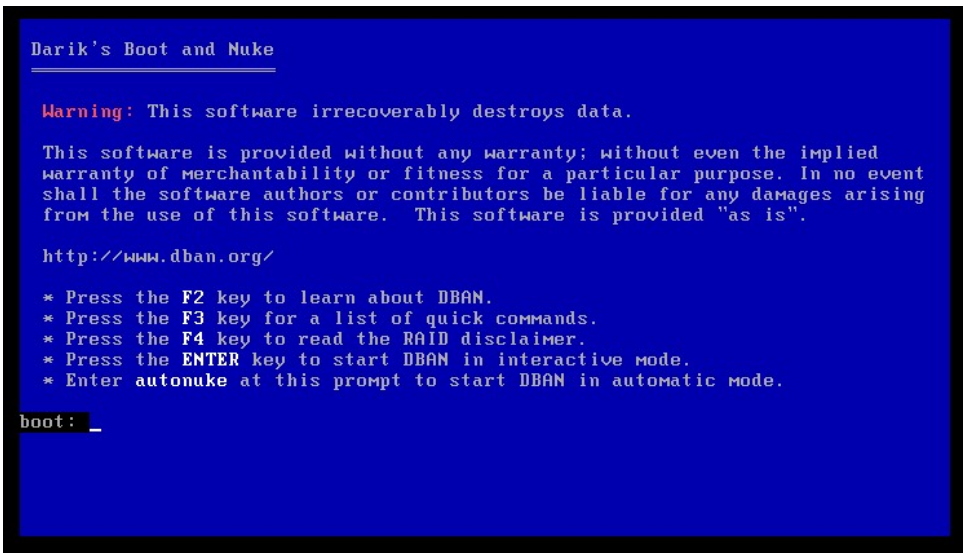
To securely wipe entire hard disks, you can use the “DBAN” tool.

To securely delete single files or directories you can use the Linux tool “Wipe”.

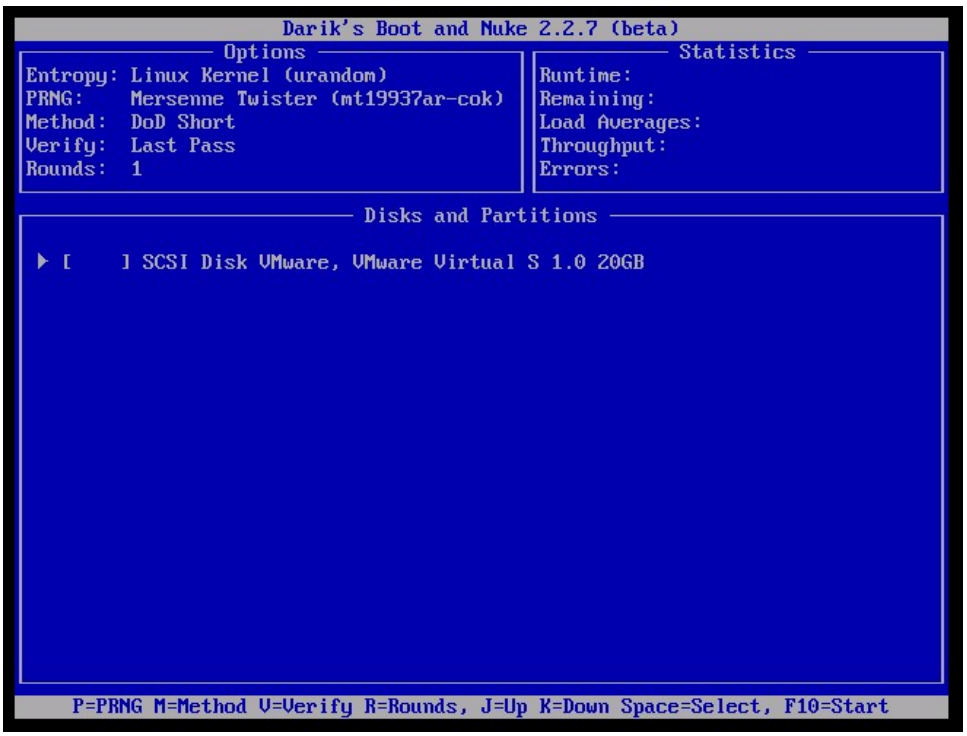
Using the DBAN tool

You can download DBAN from http://www.dban.org/.

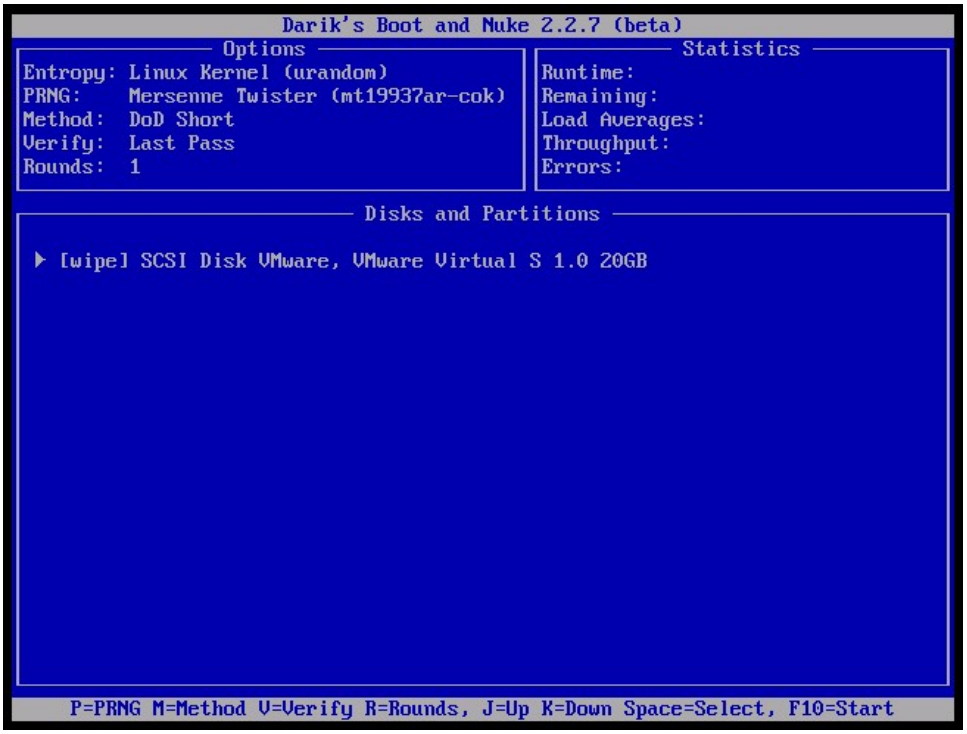
1. Create a CD with the ISO image of DBAN.
2. Boot the computer that hosts the device you want to wipe securely.
3. Press the **ENTER** key to start DBAN in interactive mode.



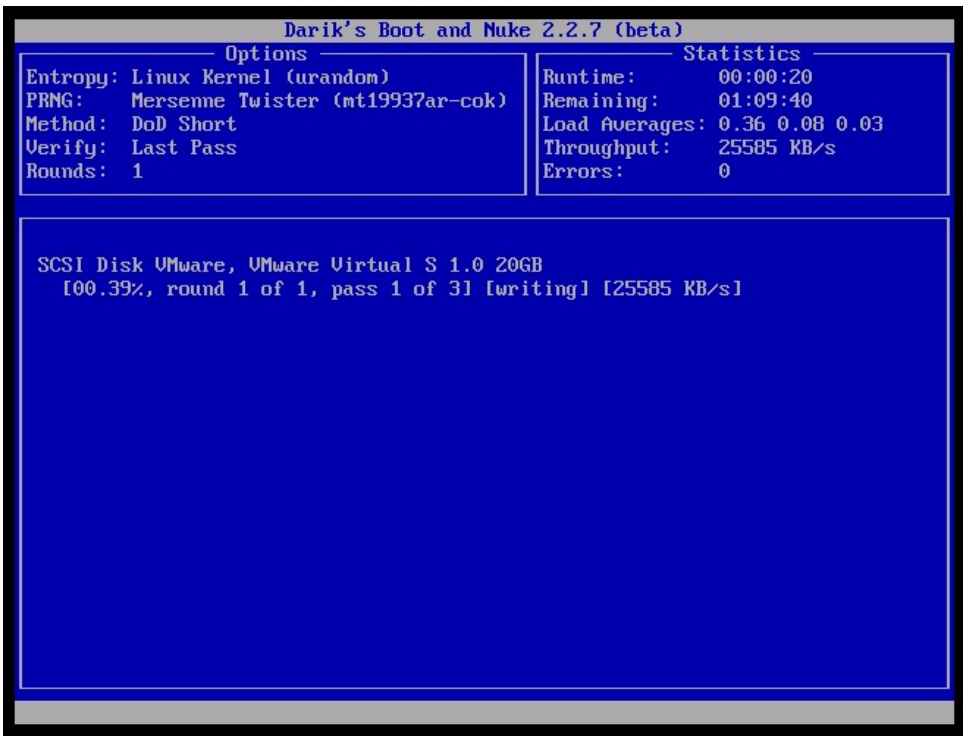
4. Type M and select the **DoD Short** method.



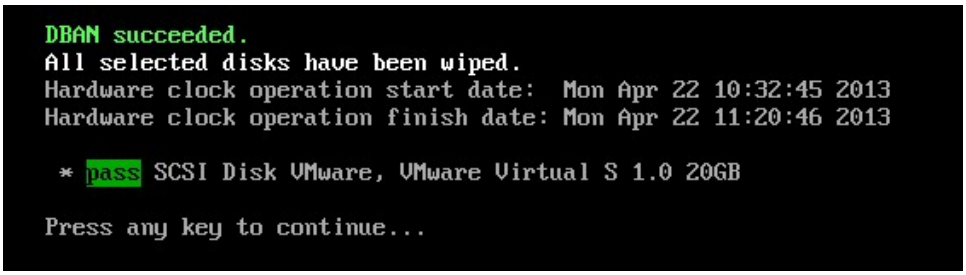
5. Select the disk or partition you want to wipe by using the up (J) and down (K) keys to move to the entry.
6. To confirm your selection, press the space bar.
7. To start wiping the disk, press F10.



8. The disk is now being wiped.



9. Make sure a dialog is displayed that confirms a successful wipe.



Using the Wipe tool

You can download Wipe from http://wipe.sourceforge.net/.

You can use the Wipe tool to securely delete single files or directories.

For example, to securely delete file *myfile.txt* run:

```
wipe -Sr -p3 myfile.txt
```

Decommission a Safer Payments instance or cluster

If a Safer Payments cluster instance or an entire Safer Payments cluster is decommissioned, you must securely delete all cardholder data by using a disk wipe tool.

Safer Payments stores cardholder data in several locations. These locations are identified and configured as described in "Configure cardholder data storage locations" on page 16.

Archived data and backups that are created by third-party applications are not in reach of the Safer Payments software itself. Therefore, they are not securely deleted automatically by Safer Payments. This aspect of this requirement must be met by organizational procedures.

PCI DSS compliance report

Safer Payments provides a built-in PCI DSS compliance report that lists all relevant configuration settings that must be changed to achieve PCI DSS compliance.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **System configuration** from the left navigation pane.

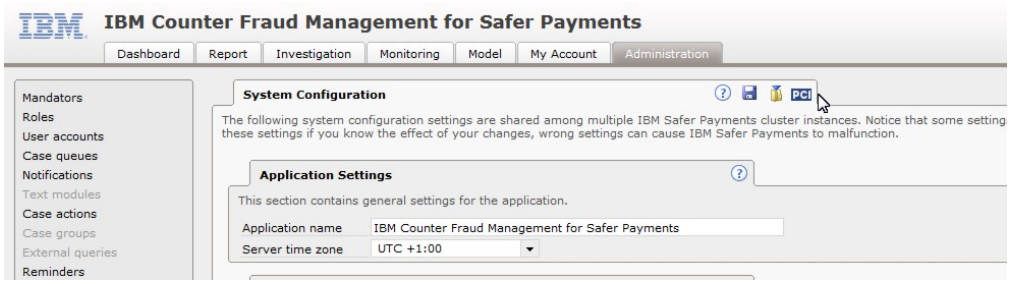


Figure 24. System configuration - Compliance Report window

3. Click the PCI icon to create the report.

The generated report lists potential issues with PCI DSS compliance for the current configuration of Safer Payments.

Use this report to configure Safer Payments according to the PCI DSS requirements. Refer to the online help for details on Safer Payments system configuration.

If you implemented all the required settings, you can rerun the report, print it and have it signed by a person responsible.

Using Safer Payments extensions

With the following extensions, Safer Payments can be used in combination with additional technologies that can be installed alongside Safer Payments on a Safer Payments server.

Security settings and logging for those software components must be set up for each of the software products in accordance to PCI DSS.

Configuration of the IBM MQ interface

Safer Payments offers the possibility to dynamically link to a local IBM MQ client to retrieve data from remote IBM MQ servers. More information on IBM MQ in general can be found in the IBM Knowledge Center. If the IBM MQ client library is installed on a local Safer Payments instance, Safer Payments loads the library at run time and uses settings in the cluster configuration to connect to remote IBM MQ servers. If no IBM MQ client is installed or loading of the IBM MQ client library fails, the function is not available.

The settings for connections to IBM MQ servers are made in **Administration > Cluster > WebSphere MQ Interface**. Each Safer Payment instance can connect to multiple queues on multiple remote IBM MQ queue managers, which are identified by a queue manager name, a target IP, and a target port. Security settings are defined in the definition of an IBM MQ channel to the queue manager. Whenever an IBM MQ connection is used to transport sensitive data over a public network, use of the "Use SSL" option is mandatory for PCI DSS compliance. Also, on the IBM MQ server side, all connection channels to a queue must be configured to use TLS 1.2 using cipher specifications listed here..

The IBM Knowledge Center article Connecting a client to a queue manager securelydescribes the process of a creating a key repository that is required for the "Use SSL" option in Safer Payments. It also describes the configurations necessary on the server side to make sure authentication using the key repository is being enforced.

IBM MQ is a product developed independently from Safer Payments, it cannot be guaranteed that the provided configuration options are always sufficient. Therefore, it is necessary to constantly monitor the IBM MQ documentation for changes to the software and the security of the used cipher suites for potential security leaks.

Configuration of a custom parser library

Safer Payments offers the possibility to dynamically link to a local library that encapsulates custom parser functionality. As the source code within this library is user defined, it is not part of Safer Payments. To use a custom library in accordance with PCI DSS, its code needs to be developed and audited separately. If no custom parser library is installed and linked to the Safer Payments library path on the Safer Payments server, the functionality is not available.

Configuration of SSO using Kerberos

Safer Payments allows SSO login using Kerberos, which needs additional setup steps on the Safer Payments server, outside of the Safer Payments configuration:

- Your Safer Payments configuration must be connected to an existing LDAP (or Active Directory) server. After turning on LDAP in Administration, System Configuration you can select the 'Allow Single Sign On' option.
- You must create a keytab file on your Kerberos (or Active Directory) server and deploy it to all Safer Payments servers that are used for API access.
- You must alter some system configuration files on the Safer Payments server to point to your Kerberos (or Active Directory) server.
- Finally, every user must run a setup step on the web browser to allow the browser to pass the users authentication parameters to the server.
- Detailed information on the setup process is available in the Safer Payments online help under Administration, System Configuration, LDAP.

Note: SSO is not required for operation in accordance to PA-DSS.

Chapter 2. Key management procedures for cryptographic keys

This section describes the cryptographic keys that are used by Safer Payments, how keys are generated, and how to enter and activate keys.

Cryptographic keys used by Safer Payments

Safer Payments encryption uses keys that are generated onsite by the Safer Payments user with the assistance of the **Keygen** program , which is provided as part of the Safer Payments software delivery.

Keygen uses master public keys to encrypt a random generated master key from which in subsequent steps any number of usage keys are generated. The master public key consists of two arbitrary passphrases of arbitrary length that are chosen by two master key holders. The encrypted master key is generated as a file that must be stored in a safe location.

When new usage keys are generated, **Keygen** is called with the encrypted master key. The master key is decrypted by the two master key holders who enter their passphrases. Now the entry of two usage public keys creates a usage key triplet. The usage key triplet consists of two arbitrary passphrases of arbitrary length that are chosen by two usage key holders.

Each usage key triplet consists of

- One usage private triplet subkey that is manually distributed to all instances of a Safer Payments cluster by the administrator.
- One left public subkey that is known only to one usage key holder.
- One right public subkey that is know only to another usage key holder.

To activate a usage key triplet, the Safer Payments instance must have the usage private key available locally. The two public keys must be available either locally entered by the usage key holders, or received from another Safer Payments instance of the cluster. The private triplet subkeys are never transmitted between the Safer Payments instances. Therefore, the parts of a key are never located on the same medium.

Safer Payments can keep multiple active and non-active key triplets in the key management function, and can switch between the active ones. A non-active triplet would be one where a subkey is not provided yet. While only one of the key triplets can be active at a time, it makes no difference, which of the key triplets is the active one.

Note: Generally, access to keys must be limited to the fewest number of custodians that are necessary. Also, keys must be stored securely in the fewest possible locations and forms. These are organizational duties to be met by the licensee.

Note: Key triplets are differentiated by their number.

Key generation

The following sections describe the principles of generating a master key, usage key triplets, and the actual key generation procedure in detail.

Master key generation process

Figure 25 shows the computational actions that are involved in master key generation.

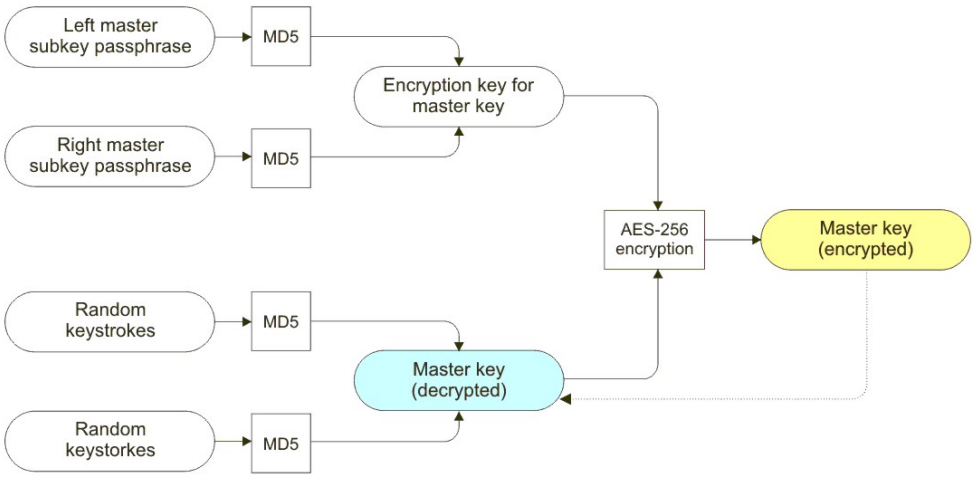


Figure 25. Master key generation process

The master key that is used by Safer Payments to encrypt and decrypt data is generated by two sets of random keystrokes that are hashed by MD5, creating a 256-bit length root key. This master key is never stored or made accessible to users. Rather, using the two passphrases of the key holders, the master key is encrypted with the AES-256 algorithm.

Important: Using the two passphrases, the encrypted master key can be decrypted. This is illustrated in Figure 25 with the dotted line.

The encrypted master key is stored in a safe place and is used, together with the passphrases of the key holders, to create the usage key triplets. The usage key triplets are the only keys that are used during Safer Payments operations.

This is also the reason why the key generator is provided as a separate utility program rather than a part of Safer Payments. Not even the encrypted master key must ever be stored on the Safer Payments server host. Use a different computer to create the encrypted master key, store it in a safe place, and generate usage key triplets whenever needed.

Usage key triplet generation process

The usage key triplet generation requires the left and right master key passphrases, and thus the presence of the key holders. Two key holders for the two public subkeys of each usage key triplet are also required. The key holders can be the same persons.

Figure 26 illustrates the process.

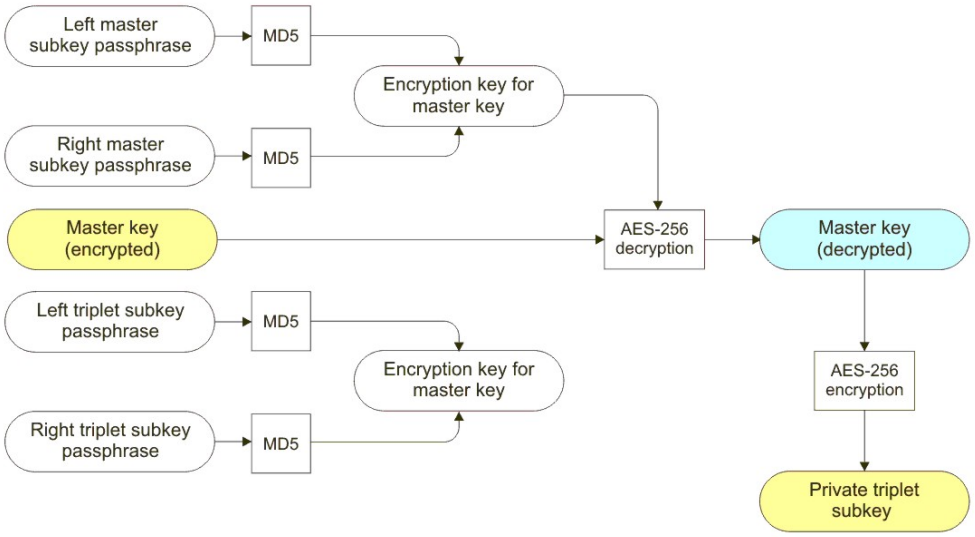


Figure 26. Private triplet subkey generation process

The encrypted master key is read from file and using the two master passphrases is decrypted in main memory only. From this decrypted version of the master key each usage key triplet is generated by encrypting the master key with a new pair of passphrases.

The result of this process is the private triplet subkey, which must be stored in the `key` directory of the Safer Payments installation. Because the file system of the Safer Payments server host is a protected area, this provides an added level of security.

A good key generation practice is to generate a number of usage key triplets in advance and then use them when they are needed.

Important: Safer Payments can reconstruct the master key in main memory from each private triplet subkey, using the two public subkeys for decryption.

Key generation steps

This section describes the key generation procedure step-by-step.

Key generation is conducted outside of Safer Payments with the **`keygen`** tool.

In summary

1. You must generate master keys.
2. The master keys are stored at a safe place and are never used by the Safer Payments software.
3. The master keys are used to generate usage keys and an empty no-fly list.
4. Only usage keys and the no-fly list are used by the Safer Payments software.
5. If you want to obtain a PA-DSS certification at a future date, keep in mind that any storage media that is used to store or distribute keys is in scope of PA-DSS requirement 2.5.2.

6. When the storage media is not required anymore, it must be securely wiped, or destroyed. See “Using a secure wipe tool” on page 25 for details.
7. You must protect and store all keys securely.

Prerequisites

Use a separate PC that is not connected to the internet to generate keys. To not block a complete PC for the occasional key generation process, you can use a PC that is started from an OS boot CD. This has the advantage that even if you disconnect the PC temporarily from the internet, no malware could have logged any of your data.

Note: You can use both Windows 64-bit and RHEL/CentOS 64-bit OS.

Obtain key generator

Keygen is now provided as part of a Safer Payments installation and is located in `/usr/bin/keygen`. Its integrity is checked when you download and verify the installation image.

Copy the contents to a portable memory location. This can be a memory card or USB stick.

Note: If you require a keygen executable for Windows, contact your support representative. To check its integrity, you can run the SHA256 Checksum Utility. It is available from https://kanguru.zendesk.com/entries/21747773-SHA256-Checksum-Utility.

Generate master key

This topic describes how to generate the master key.

To generate the master key, run the following command from the console:

`keygen master <masterkeypath> <tripletkeypath> <master_key_id>`

- *masterkeypath* is the location on your portable memory device where you want to store the master key.
- *tripletkeypath* is the location on your portable memory device where you want to store the triplet keys. The triplet keys are later physically distributed to the Safer Payments instances.
- *master_key_id* is the numeric ID for the new generated master key. Every master key that is used by your Safer Payments installation must have its unique ID.

The key generator guides you through the process of generating a master key. You need two master key holders for this process and the `masterkeypath` and `tripletkeypath` subdirectories must exist.

The master key is stored as `masterkeypath/master_key_private_<master_key_id>.iris`” and is created together with `tripletkeypath/revoked_keys.iris`.

The file `revoked_keys.iris` is used during the operation of Safer Payments to store a no-fly list of keys that Safer Payments must never use. To verify authenticity of the `revoked_keys.iris` file, it must be generated together with the initial master key.

Note: The file `revoked_keys.iris` is distributed with the initial key distribution to the Safer Payments instances. The file `master_key_private_<master_key_id>.iris`

must never be distributed to Safer Payments instances, or anywhere outside the portable memory device location. Never replace an existing "revoked_keys.iris" file in the key folder of your configuration. If you change to a usage key from another master key by the Safer Payments user interface, revoked_keys.iris is reencrypted as well.

If the two master key holders activate the master key that you generated, you can generate any number of usage keys.

You can now directly proceed to "Generate usage key triplets," or shut down the PC and store the portable memory device at a safe place until you need to generate usage keys.

Generate usage key triplets

This topic describes how to generate the usage key triplets.

To generate the usage key triplets, run the following command from the console:

```
keygen triplet <masterkeypath> <tripletkeypath>
```

- *masterkeypath* is the file location of your master key. This location must include the file name of the master key.
- *tripletkeypath* is the location on your portable memory device where you want to store the usage key triplets. The usage key triplets are later physically distributed to the Safer Payments instances. When the first key id is specified keys are not generated in *tripletkeypath*.

The key generator guides you through the process of generating a usage key triplet. You need two master key holders and two usage key holders for this process.

Keygen generates the file tripletkeypath/key_<usage_key_id>.sp.

You can repeat this process at any time to generate the number of usage key triplets that you need. The master key holder passphrases do not have to be entered for each usage key triplet generation, unless you quit the key generator.

If you generated all the usage key triplets you need, shut down the PC and store the portable memory device at a safe place until you need to generate more usage keys.

Distribute keys

This topic describes how to distribute the keys.

All the key_*n*.iris files (private triplet subkeys) that you want to use with your Safer Payments installation, must be copied manually from the portable memory device to the key subdirectories of all Safer Payments instances.

If you copy usage key triplets to running Safer Payments instances, you must reload the keys as described in "Activate data encryption - step 2" on page 29. Safer Payments reloads keys automatically whenever it restarts. Do not overwrite or replace the revoked_keys.iris or the key_*n*.iris files in the key subdirectory.

The first time that you distribute keys to Safer Payments instances, you must include the file revoked_keys.iris that was generated during the initial creation of the master key. This file stores the no-fly list of revoked keys. Never overwrite this file manually once it is delivered to the Safer Payments instances. Make sure that

this file is writable for Safer Payments to revoke keys or to reencrypt the file. For example, if you change to another master key.

The content of the encrypted `revoked_keys.iris` files might differ on each instance after you reencrypt or revoke a key. As the encryption of this file adds a random token, the encrypted result differs on each instance. Nevertheless, the stored no-fly list is always the same.

When you copy the files to the `key` subdirectories, make sure that you adjust the user and group access privileges so that only the Safer Payments process user can access those files.

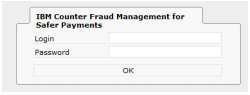
Leave a copy of the usage key triplet files on the portable memory device so that you have a reference of generated keys. You must protect and store the device securely.

Activate keys

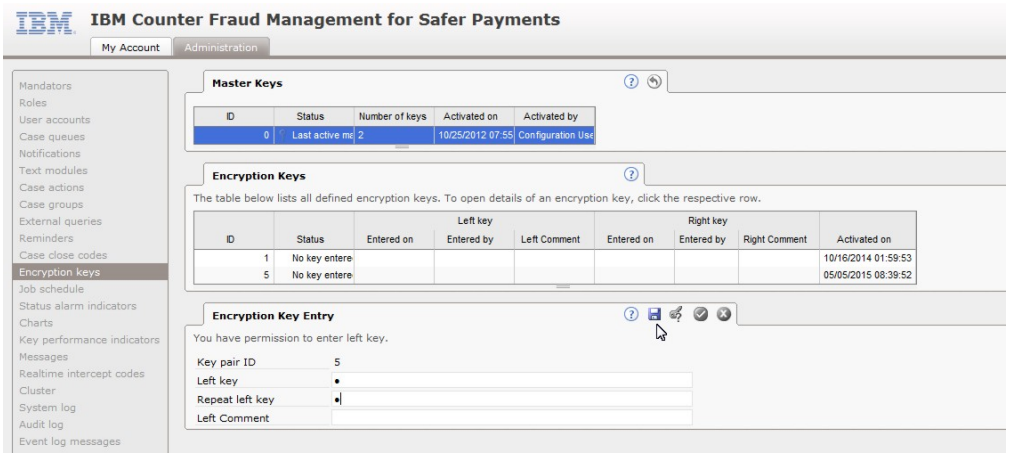
This topic describes how to activate encryption keys.

To activate a usage key triplet, the two passphrases must be entered into the Safer Payments user interface.

1. Assuming you are the left key holder, log in.



2. The Safer Payments user interface opens and displays **Encryption keys** window.
3. In the table, click the row of the key instance you want to activate.



4. In the **Left key** field, enter your key and repeat it for verification.
5. Click the  icon.
6. The right key holder must also log in and follow these steps.
7. The user who has the global privilege to activate key triplets must log in and go to **Administration > Encryption keys**

Figure 27. Activate Encryption Keys

Note: The global privilege to activate usage key triplets can be granted to the key holders or any other user.

8. In the **Encryption Key Entry** window, click the  icon.

You can prepare more than one key for activation, and users with respective privileges can switch between them by activating a key.

If a key is revoked, the key file is automatically securely erased on all Safer Payments instances in a cluster. The revoked key is also added to the no-fly list to ensure that this key cannot be active again in Safer Payments.

Safer Payments instances in a cluster share the passphrases over their encrypted network connection (ECI). The private triplet subkey of the usage key triplet is transferred manually by the operator. Therefore, the private key and the public keys never travel together on the same medium. Thus, spying out only one of the channels does not deliver sufficient information to decrypt Safer Payments.

Because Safer Payments instances share the public keys, the key holders do not have to enter them each time a Safer Payments instance is started. If one Safer Payments instance is still running in the cluster, passphrases do not have to be reentered. Only when you start the first Safer Payments instance, passphrases must be entered.

You can simultaneously start all Safer Payments instances because in key-entry mode the user interface is partially active to allow for key entry. When keys are entered on any Safer Payments instance of the cluster, they are shared within the cluster and the Safer Payments instances start. This might take a few minutes.

Note: A key is automatically deactivated, if you activate another key.

Precautions and possible errors

- If you use a Flash-based portable memory device, which most USB sticks or SD cards are, it is difficult to securely erase data from them. Therefore, you must store the portable memory device in a safe location for the entire duration of the master key being valid. If you ever need to erase the master key on such a portable memory device, the safest way is physical destruction.
- If Safer Payments cannot locate the `revoked_keys.sp` file during startup, or if the file is tampered with, Safer Payments creates a log message and shuts down immediately.

- If Safer Payments finds an active key that is on the no-fly list, Safer Payments securely deletes the key from the `key` subdirectory and shuts down immediately. If the key is not active, Safer Payments creates a log message, securely deletes the key from the `key` subdirectory, and continues with startup.
- If you run a key reload from the Encryption Keys window of the Safer Payments interface, the following problems can occur:
  - If Safer Payments cannot locate the `revoked_keys.sp` file, or the file is tampered with, an error message on the user interface and a log message are created, reloading is stopped, yet operations resume.
  - If Safer Payments finds keys that are on the no-fly list, the keys are securely deleted from the `key` subdirectory, an error message on the user interface and a log message are created.

Enforce regular key changes

This topic describes how to define regular key changes and how to set key life alerts.

Regular key changes are recommended.

The National Institute of Standards and Technology has developed recommendations for key management. Their guidelines assist you in defining the correct key retention periods for your organization.

You can download the *NIST Special Publication 800-57* here: http://csrc.nist.gov/publications/PubsSPs.html#SP%20800

Based on this, we recommend a maximum key life of 120 days, and a maximum master key life of three years.

Important: Retirement or replacement of keys is required, if the integrity of the key has been weakened, or keys are suspected of being compromised.

Define maximum key life

You can define the maximum key life and the maximum master key live as follows.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **System configuration** from the left navigation pane.
3. Scroll down the **System Configuration** window to locate the window titled **Encryption**.

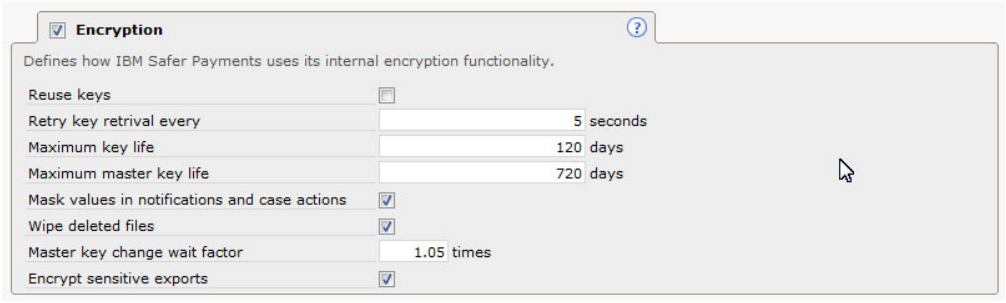


Figure 28. Encryption window

4. In the **Maximum key life** field, enter the number of days you defined in your organization.

If the maximum key life is reached and no key is changed during this period, Safer Payments automatically shuts down.

Set maximum key life alerts

Safer Payments provides a **Status Alarm Indicator** (SAI) that alerts, if the end of the maximum key life comes closer. SAI alerts can be sent to the Safer Payments dashboard screen, and can be distributed by email, or log messages.

You must define the following two status alarm indicators:

- One for the encryption key, it must have the alarm type **encryption key valid until**.
- One for the master key, it must have the alarm type **master key valid until**.

1. On the Safer Payments user interface, click the **Administration** tab.
2. Select **Status alarm indicators** from the left navigation pane.
3. In the **Status Alarm Indicators** window, click the  icon to create a new status alarm indicator.

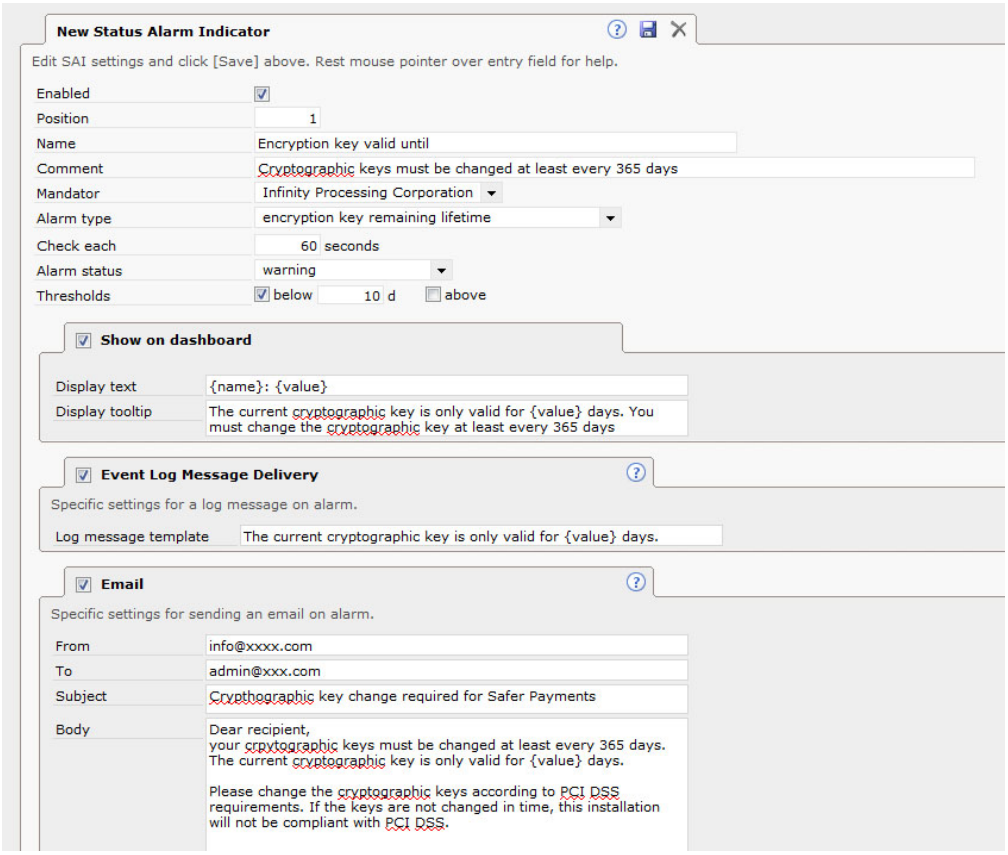


Figure 29. New Status Alarm Indicator window

Figure 29 shows an exemplary SAI definition for monitoring the last encryption key change.

This SAI assumes a maximum key life of 120 days. If the current key is valid for only 10 more days a warning is displayed on the dashboard, a mail is sent out, and a log message is created.

Revoke Keys

Authorized users can revoke cryptographic keys in the **Encryption Key Entry** window (Figure 27 on page 51) by clicking the  icon. If a key is revoked, Safer Payments securely deletes the usage private key stored on disk, and removes the two usage public keys from main memory in all cluster instances.

Note: Only inactive keys can be revoked.

However, you must manually delete the revoked keys from all other storage locations using a secure wipe tool. For example, the media used for key distribution. See "Using a secure wipe tool" on page 25for details.

Change the master key

Carefully consider when to change the master key. During the change of the master key, all cluster instances become inactive.

While it is still possible to score transactions, you cannot change the configuration or investigate cases during the change process. The change affects all data that is stored in Safer Payments, which means such a change process can take several hours to complete.

Changing the master key requires the global privilege to change the master key. This privilege must be granted to the user in advance.

1. On the Safer Payments user interface, click the **My account** tab.
2. Scroll down to the **Global Privileges** window.
3. In the **Key Management** field, select **activate and revoke keys and view encryption management, and change master key** from the pull-down menu.
4. Save your changes.

The process to change the master key is as follows:

1. Generate a new master key with the **keygen** tool with a new master key ID. See "Generate master key" on page 48 for details.
2. Generate new private keys from the new master key. See "Generate usage key triplets" on page 49 for details.
3. Copy the new private keys (`key_<key_id_n>.iris`) into the `key` folder on all instances
4. Do not replace any file of the `key` folder while copying.
5. Click the **Administration** tab.
6. Select **Encryption keys** from the left navigation pane.
7. Click the  icon to reload private keys from disk.
8. The new master key is displayed. Log out.
9. The left key holder must log in and insert the left key.
10. The right key holder must log in and insert the right key.
11. Log in. You must have the right to activate a key.

12. Click the  icon to change the master key.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those

websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Terms and Conditions for Product Documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein. IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as

determined by IBM, the above instructions are not being properly followed. You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/VSE enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Index

T

U

V

Readers' Comments — We'd Like to Hear from You

IBM Counter Fraud Management for Safer Payments 5.6.0
Implementation Guide
Version 5 Release 6

Publication No. SC34-2787-00

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Submit your comments using one of these channels:

- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: +49-7031-163456
- Send your comments via email to: s390id@de.ibm.com
- Send a note from the web page: http://www.ibm.com/software/products/en/counter-fraud-management-safer-payments

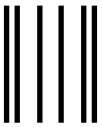
If you would like a response from IBM, please fill in the following information:

Name	Address
Company or Organization	
Phone No.	Email address

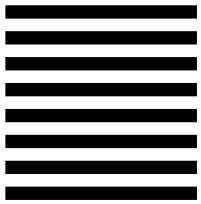
Readers' Comments — We'd Like to Hear from You
SC34-2787-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Deutschland Research & Development GmbH
Department 3282
Schoenaicher Strasse 220
71032 Boeblingen
Germany

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Printed in USA

SC34-2787-00