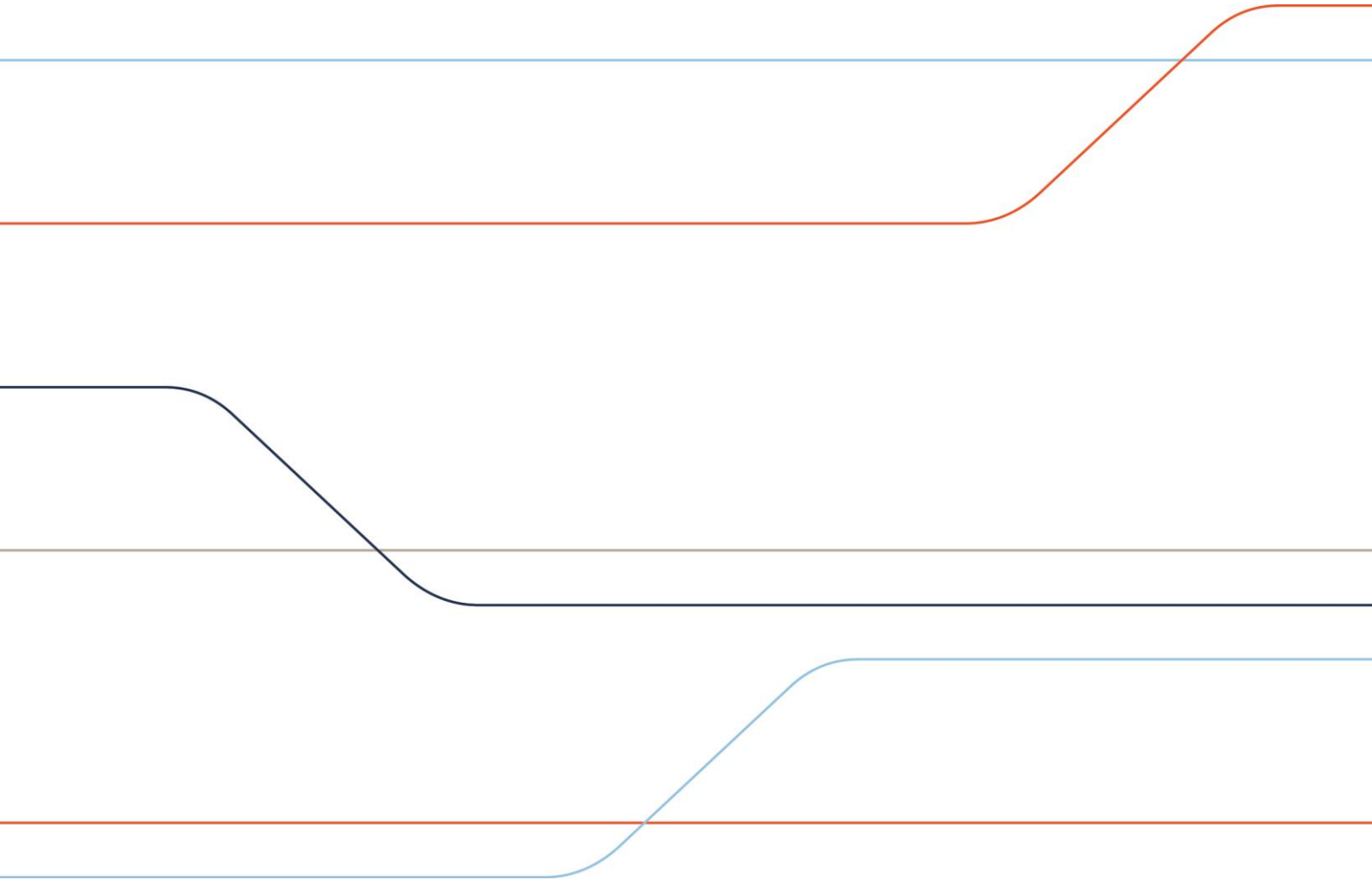


Payment Page

Spécifications

Version 5.1



1	Introduction	4
1.1	Conditions préalables	4
1.2	Affichage de la Payment Page	4
1.3	Sécurité des données et PCI DSS	5
1.4	Moyens de paiements pris en charge	6
1.5	Format des données	6
2	Saferpay Client Library	7
2.1	Conditions	7
2.1.1	.NET Client Library.....	7
2.1.2	Java Client Library	7
2.2	Installation	7
2.2.1	.NET Client Library.....	7
2.2.2	Java Client Library	7
2.3	Configuration du serveur Proxy	8
2.3.1	.NET Client Library.....	8
2.3.2	Java Client Library	9
2.4	Génération de clés	10
2.4.1	.NET Client Library.....	10
2.4.2	Java Client Library	10
3	Classes et méthodes de la Client Library	11
3.1	Résumé	11
3.1.1	Création du lien de paiement	11
3.1.2	Vérification de la demande de paiement	11
3.1.3	Comptabilisation d'une réservation	11
3.2	Classe MessageFactory	11
3.3	Classe MessageObject	12
3.4	Méthode Open()	12
3.5	Méthode CreatePayInit()	12
3.6	Méthode VerifyPayConfirm()	12
3.7	Méthode CreateRequest()	12
3.8	Méthode SetAttribute()	12
3.9	Méthode GetAttribute()	12
3.10	Méthode GetPostURL()	12
3.11	Méthode GetPostData()	12
3.12	Méthode GetPostSignature()	13
3.13	Méthode Capture ()	13
4	Saferpayhttps Interface	14
4.1	Adresses de la https Interface	14
4.2	Déroulement d'une transaction	14
4.3	Création de l'URL de paiement via l'adresse CreatePayInit	15
4.4	Vérification de la réponse d'autorisation via l'adresse VerifyPayConfirm	15
4.5	Exemple de VerifyPayConfirm	16
4.6	Comptabilisation d'un paiement via l'adresse PayComplete	16
4.7	Exécution de la Payment Page via l'adresse Redirect	17
4.7.1	Inconvénients de Redirect.....	17
4.7.2	Adresse https Interface	17
4.7.3	Déroulement de la transaction avec Redirect.....	17
5	Etapes du traitement	18
5.1	Résumé	18

5.2	Description de la procédure	18
6	Paramètres	20
6.1	Paramètres PayInit	20
6.1.1	Codes langues pour LANGID	26
6.1.2	ID des moyens de paiement pour - PAYMENTMETHODS	26
6.1.3	Paramètres PayConfirm de la réponse d'autorisation	27
6.1.4	Paramètres d'adresse lors de l'utilisation de Masterpass	29
6.2	Demande VerifyPayConfirm	30
6.3	Demande PayComplete	31
6.4	Réponse PayComplete	32
7	Adaptation avec feuilles de styles en cascade	33
7.1	Variantes de style	33
7.1.1	Modifier la Payment Page mobile de Saferpay	33
7.1.2	Présentation totalement nouvelle pour la Payment Page	33
7.2	Taille de l'Inline Frame	34
7.3	Utilisation du CSS	34
7.3.1	Element Name	34
7.3.2	Class Name	34
7.3.3	Element ID	34
7.3.4	Element Attribute	34
7.3.5	Sélecteurs CSS	35
7.4	Informations importantes pour l'utilisation de CSS	35
7.5	Noms de catégories CSS	35
7.5.1	Catégories générales	35
7.5.2	Spécifique pour Payment Selection	37
7.5.3	Spécifique pour formulaire de cartes	37
7.5.4	Spécifique pour recouvrement direct	37
7.5.5	Spécifique pour Online Banking	38
7.5.6	Spécifique pour formulaire de facture	38
7.5.7	Spécifique pour formulaire d'adresse (y compris formulaire d'adresse Billpay)	38
7.5.8	Spécifique pour CG	38
7.5.9	Spécifique pour pages Redirect (y compris 3D-Secure)	38
7.5.10	Spécifique pour pages d'erreur	38
7.5.11	Spécifique pour Confirmation Page	38
8	Environnement de test Saferpay	39
9	Exemples	40
9.1	Remarque importante	40
9.2	C# avec la .NET LIB	40
9.3	Java avec la Java LIB	41
9.4	Exécution d'une ligne de commande avec la Java LIB	42
9.5	https Interface	44
10	Codes d'erreur	46
11	Contact	47
11.1	Saferpay Integration Team	47
11.2	Saferpay Support Team	47

1 Introduction

La Saferpay Payment Page, par la suite abrégée en PP, est un formulaire de paiement en ligne mis à disposition par Saferpay. Ce document décrit l'intégration de la PP dans les systèmes actuels de boutique en ligne avec la Saferpay Client Library et la Saferpay https Interface. Pour simplifier, la Saferpay Client Library est abrégée dans ce document LIB et la Saferpay https Interface en HI.

1.1 Conditions préalables

L'utilisation de la PP nécessite les conditions suivantes :

- Une licence conforme et par conséquent l'existence d'un identifiant valide composé d'un nom d'utilisateur et d'un mot de passe pour le Saferpay System.
- Un terminal Saferpay actif et disponible au minimum, sur lequel les paiements peuvent être effectués, ainsi que l'existence du Saferpay TERMINALID ou du Saferpay ACCOUNTID correspondant.
- Un contrat d'acceptation pour les cartes de crédit ou tout autre moyen de paiement disponible et valide.
- Afin de pouvoir utiliser la Saferpay HI, les informations du commerçant sur la HI doivent être paramétrées. Une fois l'enregistrement effectué, les certificats en vue d'une communication encodée selon la norme SSL sont édités pour le commerçant et mis à disposition par Saferpay. L'enregistrement est gratuit mais doit être demandé cependant pour chaque Saferpay Account. Pour cela, veuillez envoyer un courriel comportant uniquement votre demande d'enregistrement brute à onlinepayment@six-group.com si vous avez conclu votre contrat avec Saferpay en Suisse ou à service.saferpay@six-payment-services.com si vous avez conclu votre contrat dans un autre pays (D, NL, A...). Si vous utilisez l'adresse Redirect, veuillez indiquer obligatoirement à nos collègues l'adresse Internet complète pour le SUCCESSLINK, car elle doit être ensuite déterminée une fois pour toute chez Saferpay.

1.2 Affichage de la Payment Page

Outre l'affichage complet de la Payment Page, Saferpay supporte également l'intégration de celle-ci dans un Inline Frame (iframe). L'application ne permet pas l'ouverture de la PP comme Pop-Up.

Même si Saferpay permet l'affichage de la PP dans un iframe, nous recommandons l'affichage complet. Selon nous, il améliore notablement la convivialité pour le client, car l'URL et le certificat SSL sont visibles à tout moment. Le client voit ainsi clairement chez quel fournisseur, c'est-à-dire sur quel site Internet, il se trouve. Sans compter que cela réduit les annulations de paiement pour cause de questions de sécurité du prestataire.



PayPal bloque les demandes de paiement à partir d'un iframe!

Pour cette raison, dans le cas de PayPal, la Payment Page Saferpay provoque une échappatoire de l'iFrame et démarre ensuite PayPal sur une page entière. Ainsi, le démarrage de la SUCCESS-Page n'a plus lieu dans l'iFrame mais sur une page entière. Nous vous prions de veiller à cela lors de l'intégration de PayPal.

Il n'est pas exclu que d'autres fournisseurs de moyens de paiement et exploitants de pages d'authentification pour 3-D Secure PayPal fassent de même. Aux fins de sécurité, demandez à votre prestataire de moyens de paiement s'il autorise l'affichage de la PP dans un iframe ou si, pour des raisons de sécurité, il interdit l'intégration dans un iframe!

1.3 Sécurité des données et PCI DSS

Les établissements de cartes de crédit ont mis en place le programme de sécurité PCI DSS (Payment Card Industry Data Security Standard), afin de lutter contre les fraudes liées aux cartes de crédit et à leur usage abusif.

Lors de l'utilisation de la PP, le titulaire de la carte saisit son numéro de carte de crédit et la date d'expiration non pas sur l'application de commerce en ligne du commerçant mais dans la fenêtre de paiement Saferpay (PP). Comme l'application de commerce en ligne et Saferpay sont gérés sur des plates-formes physiques séparées, il n'y a aucun risque que les données des cartes de crédit puissent être enregistrées dans la banque de donnée du système du commerçant. Le risque d'usage abusif des données de la carte de crédit est ainsi réduit et, pour le commerçant, cela réduit nettement le temps d'acquisition du certificat PCI DSS.

Votre acquirer ou une entreprise spécialisée peut répondre à vos questions concernant la norme PCI DSS (voir <https://www.pcisecuritystandards.org>).

1.4 Moyens de paiements pris en charge

La SaferpayPayment Page peut à l'heure actuelle traiter les moyens de paiement suivants :

- Visa (avec la technologie de sécurité 3-D Secure « Verified by Visa »)
- MasterCard (avec la technologie de sécurité 3-D Secure « MasterCard SecureCode »)
- Maestro international
- V PAY
- American Express
- Diners Club
- J.C.B.
- Système de recouvrement direct électronique ELV (en Allemagne uniquement)
- giropay
- iDEAL
- PayPal
- eps
- PostFinance Card et PostFinance E-Finance
- Przelewy Online
- MasterPass *

** Pas un moyen de paiement au sens propre. En fait, il s'agit d'un portefeuille électronique (wallet), qui contient les données des moyens de paiement du client.*

1.5 Format des données

Les abréviations suivantes sont utilisées dans ce document pour le format des données :

- a lettres (a - z, A - Z)
- n signes numériques (0 - 9)
- an signes alphanumériques (a - z, A - Z, 0 - 9)
- s caractères spéciaux (:? ,-(+'.)/ et espace)
- ans signes alphanumériques et caractères spéciaux

2 Saferpay Client Library

La Saferpay LIB est installée sur le serveur qui met à disposition l'application du commerçant. Une fois l'installation effectuée, les classes et méthodes Saferpay sont disponibles sur le serveur.

Des droits d'utilisateur root ou d'administrateur sur le serveur cible sont nécessaires pour l'installation de la LIB et la création de clés qui éventuellement en découle!

La LIB existe en version .NET- ou Java. Les fichiers correspondant peuvent être téléchargés dans la section téléchargement du Saferpay Backoffice a l'adresse ci-dessous :

<https://www.saferpay.com/download/>

S'il advient que ni .NET LIB, ni Java LIB ne peuvent être utilisées ou qu'une installation locale n'est pas possible, la Saferpay https Interface est une alternative mise à disposition.

2.1 Conditions

2.1.1 .NET Client Library

Comme la Saferpay .Net Client LIB a été compilée dans le .Net Framework 2.0, celui-ci doit également être installé sur le serveur cible.

2.1.2 Java Client Library

Sur le serveur cible, un Sun Java Runtime Environment (JRE) à partir de la version 1.3.1 ou plus récente doit être installé. Les environnements alternatifs comme OpenJDK ou d'IBM ne sont pas utilisables, à moins de complications supplémentaires.

2.2 Installation

2.2.1 .NET Client Library

Lancez le fichier d'installation téléchargé « saferpay_dotnet.exe » et suivez les indications de l'assistant d'installation.

2.2.2 Java Client Library

Dézippez le fichier zip téléchargé « saferpay_java.zip » et copiez le fichier contenu « Saferpay.jar » dans le dossier jre/lib/ext en vue de l'intégration dans Java.

Pour l'intégration dans un autre langage de programmation ou de script, le fichier « Saferpay.jar » peut être copié dans n'importe quel dossier.

2.3 Configuration du serveur Proxy

Si la communication dans le réseau est effectuée à partir d'un serveur Proxy, des données de configuration correspondantes sont nécessaires pour les LIB Saferpay.

2.3.1 .NET Client Library

Pour utiliser un serveur Proxy, un paramètre doit être ajouté au fichier « config.xml ». Le fichier se trouve dans le dossier d'installation du .NET Client, par exemple dans
C:\Programme\Saferpay\Client\.

Serveur Proxy avec reconnaissance de l'utilisateur

Pour communiquer à partir d'un Proxy avec des données d'accès individuelles, les paramètres suivants du « config.xml » doivent être ajoutés ; l'ordre n'a ici aucune importance :

```
PROXYPASSWORD="geheim"  
PROXYUSERNAME="MyProxyUser"  
PROXYADDRESS="http://localhost:8080"  
USEPROXY="True"  
USEDEFAULTCREDENTIALS="False"
```

Serveur Proxy sans reconnaissance de l'utilisateur

Pour communiquer sans saisir l'utilisateur et le mot de passe Proxy, les paramètres suivants du « config.xml » doivent être ajoutés ;

```
PROXYADDRESS="http://localhost:8080"  
USEPROXY="True"  
USEDEFAULTCREDENTIALS="True"
```

Selon la configuration du Proxy, le contenu de « config.xml » apparaît alors ainsi :

```
<IDP MSGTYPE="SetupResponse" GXID="6216B171-B449-4D02-A114-D42AB58D42AE"  
CUSTOMERID="99867" VERSION="47"  
VTAUTOURL="https://www.saferpay.com/user/setup.asp"  
VTURL="https://www.saferpay.com/vt2/Pay.aspx" VTKEYID="1-0"  
CAPTUREURL="https://www.saferpay.com/scai2/index.aspx"  
VTSCRIPTURL="http://www.saferpay.com/OpenSaferpayScript.asp"  
USEDEFAULTCREDENTIALS="True" USEPROXY="True"  
PROXYADDRESS="http://localhost:8080" />
```


2.3.2 Java Client Library

Sur la LIB Java, la configuration d'un serveur Proxy peut être effectuée à partir d'un fichier « settings.xml » ou à partir d'une exécution de lignes de commande. Si vous utilisez « settings.xml », celle-ci doit être créée dans le même dossier contenant « saferpay.jar », par exemple dans jre/lib/ext.

Serveur Proxy avec reconnaissance de l'utilisateur

Exemple « settings.xml » :

```
<IDP PROXYHOST="10.23.209.100" PROXYPORT="8080" PROXYUSERNAME="myUserId"
PROXYPASSWORD="myPassword" TRACEOPT="rawhttp" VERSION="1" USEPROXY="1" />
```

Exemple Exécution de lignes de commande :

```
--proxyHost 10.23.209.100 --proxyPort 8080 --proxyUser myUserId
--proxyPassword myPassword
```

Serveur Proxy sans reconnaissance de l'utilisateur

Exemple « settings.xml » :

```
<IDP PROXYHOST="10.23.209.100" PROXYPORT="8080" TRACEOPT="rawhttp"
VERSION="1" USEPROXY="1" />
```

Exemple Exécution de lignes de commande :

```
--proxyHost 10.23.209.100 --proxyPort 8080
```

2.4 Génération de clés

Outre la communication encodée par SSL entre la LIB et les serveurs Saferpay, les informations d'un compte Saferpay sont équipées et protégées par une signature numérique d'après la procédure PGP (Pretty Good Privacy). Dans ce but, une paire de clés doit être générée pour chaque Saferpay Account et enregistrée sur le serveur du commerçant. Afin de générer ces clés, un identifiant valide et le mot de passe pour le Saferpay Backoffice sont nécessaires. Une fois la création des clés réussie, le mot de passe du Saferpay Backoffice peut être modifié car les clés ne doivent être générées en tout et pour tout qu'une seule fois. Les paires de clés générées conservent leur validité sans limite de temps et doivent être par conséquent conservées avec soin et leur accès doit être contrôlé.

2.4.1 .NET Client Library

Une fois .NET LIB installée, une interface graphique est mise à disposition pour la création des clés. Elle se trouve ici :



Le Saferpay Client Setup apparaît. Suivez ensuite les indications.

2.4.2 Java Client Library

La création de clés avec la Java LIB s'effectue à l'aide d'une ligne de commande. Pour cela, allez dans le dossier contenant Saferpay.jar et saisissez la commande suivante :

```
java -jar Saferpay.jar -conf -p <dossier cible> -r  
https://www.saferpay.com/user/setup.asp -u e99867001 -w XAjc3Kna
```

L'exemple utilise les informations d'accès pour le compte -test Saferpay.

L'aide de ligne de commande apparaît après la commande suivante :

```
java -jar Saferpay.jar -h
```

3 Classes et méthodes de la Client Library

Les classes et méthodes disponibles pour l'intégration de la Saferpay Client LIB sont décrites dans cette section.

3.1 Résumé

3.1.1 Création du lien de paiement

Le lien de paiement est généré avec la méthode `CreatePayInit()`. Le Message Object créé doit être complété avec les paramètres de transaction avant l'exécution de `GetPostURL()`.

- 1) Création d'un MessageFactory Object.
- 2) Ouverture de la configuration correspondante avec `Open()`.
- 3) Exécution de `CreatePayInit()`, afin de recevoir un MessageObjectvierge.
- 4) Exécution de `SetAttribute()` avec le MessageObjectpour déterminer les paramètres.
- 5) Exécution de `GetPostURL()` pour recevoir un lien de paiement pour la Payment Page.

3.1.2 Vérification de la demande de paiement

- 1) Création d'un MessageFactory Object.
- 2) Ouverture de la configuration correspondante avec `Open()`.
- 3) Exécution de `VerifyPayConfirm()` pour la vérification de DATA et SIGNATURE.
- 4) Sélection de ID et TOKEN avec `GetAttribute()`.
- 5) Enregistrement des valeurs ID et TOKEN.

3.1.3 Comptabilisation d'une réservation

- 1) Création d'un MessageFactory Object.
- 2) Ouverture de la configuration correspondante avec `Open()`.
- 3) Exécution de `CreateRequest("PayComplete")`, afin de recevoir un MessageObject .
- 4) Exécution de `SetAttribute()` avec le MessageObject pour déterminer ID et TOKEN.
- 5) Exécution de `Capture()` avec le MessageObject.

3.2 Classe MessageFactory

```
Class MessageFactory
{
    void Open(String path);
    MessageObject CreatePayInit();
    MessageObject VerifyPayConfirm(String data, String signature);
    MessageObject CreateRequest(String msgtype);
};
```

3.3 Classe MessageObject

```
Class MessageObject
{
    void SetAttribute(String name, String value);
    String GetAttribute(String name);
    String GetPostURL();
    String GetPostData();
    String GetPostSignature();
    void Capture();
};
```

3.4 Méthode Open()

Il s'agit de la référence à la paire de clés du compte du commerçant .Open() doit être exécuté avant toutes les autres méthodes du MessageFactory Object afin que d'autres exécutions de fonctions se réfèrent à ces clés grâce à cette MessageFactory.

3.5 Méthode CreatePayInit()

Avec CreatePayInit(), un lien de paiement peut être créé. Celui-ci est généré en incorporant les paramètres déterminés avec GetURL().

3.6 Méthode VerifyPayConfirm()

Vérifie la signature numérique du message de confirmation (MSGTYPE=PayConfirm) envoyé à la boutique en ligne via SUCCESSLINK ou NOTIFYURL, afin de s'assurer que la réponse n'a pas été manipulée.

3.7 Méthode CreateRequest()

Génère un nouveau RequestMessageObject pour le type de message donné (msgtype). Avec CreateRequest("PayComplete") p. ex. une réservation peut être comptabilisé ou rejetée, une comptabilisation annulée et le bouclage journalier déclenché., Pour la comptabilisation CreateRequest("PayComplete") a toujours besoin des paramètres ID et ACCOUNTID. Pour une comptabilisation avec réduction du montant, le paramètre AMOUNT doit être soumis en supplément. Le paramètre ACTION est nécessaire pour annuler une transaction réservée ou comptabilisée ainsi que pour déclencher le bouclage journalier.

3.8 Méthode SetAttribute()

SetAttribute() détermine les paramètres nécessaires pour le message. Veuillez faire attention aux majuscules lors de la saisie des noms des paramètres.

3.9 Méthode GetAttribute()

GetAttribute() fournit en retour la valeur d'un paramètre du message. Si le paramètre ne figure pas dans le message, l'activation échoue. Veuillez faire attention aux majuscules lors de la saisie des noms des paramètres.

3.10 Méthode GetPostURL()

L'exécution de GetPostURL() fournit le lien de paiement du message.

3.11 Méthode GetPostData()

L'exécution de GetPostData() fournit le DATA du message.

3.12 Méthode GetPostSignature()

L'exécution de GetPostSignature() fournit la SIGNATURE du message

3.13 Méthode Capture ()

L'exécution de Capture() transmet le message du type de message CreatePayComplete

4 Saferpayhttps Interface

La Saferpayhttps Interface peut être utilisée comme alternative à la Saferpay Client Library. Cela peut être le cas lorsque la Saferpay LIB ne peut pas être installée ou utilisée sur le système cible.

4.1 Adresses de la https Interface

La Saferpayhttps Interface est accessible via les adresses Internet ci-dessous :

Création d'une URL de paiement :

<https://www.saferpay.com/hosting/CreatePayInit.asp>

Vérification d'une réponse d'autorisation :

<https://www.saferpay.com/hosting/VerifyPayConfirm.asp>

Comptabilisation d'un paiement :

<https://www.saferpay.com/hosting/PayCompleteV2.asp>

Pour des raisons de sécurité, les paramètres ACTION et AMOUNT ne sont pas pris en compte par l'adresse PayComplete de la HI.



Attention ! La plupart des frameworks vérifient automatiquement la validité du certificat du serveur. Toutefois, nous vous recommandons d'utiliser l'interface https de Saferpay pour vous assurer que votre application vérifie le certificat du serveur afin d'éviter toute attaque de l'homme du milieu (man in the middle attack).

4.2 Déroulement d'une transaction

Une transaction se déroule en principe sur le schéma suivant :

1. Création de l'URL de paiement (CreatePayInit)
2. Le paiement est réalisé par le client en autonomie sur Saferpay.
3. Si le paiement est réussi, le SUCCESSLINK est exécuté avec la réponse d'autorisation.
4. La réponse doit satisfaire le contrôle de plausibilité/manipulation (VerifyPayConfirm).
5. Pour finir, le paiement doit être comptabilisé afin de déclencher le transfert d'argent suite au bouclage journalier. La comptabilisation peut être effectuée manuellement via le Saferpay Backoffice ou automatiquement avec la commande PayComplete.

4.3 Création de l'URL de paiement via l'adresse CreatePayInit

Le système de la boutique en ligne transmet le paramètre PayInitSaferpay, via GET ou POST à la HI où un URL de paiement encodé est généré à l'aide de chaque signature de compte Saferpay avant d'être renvoyé au système de la boutique. L'URL de paiement ainsi généré peut ensuite être utilisé pour le paiement lorsqu'il est inséré à la page de la boutique en ligne par exemple sous la forme d'un lien ou d'un bouton.

Exemple d'exécution (GET):

```
https://www.saferpay.com/hosting/CreatePayInit.asp? AMOUNT=...
```

Le résultat est renvoyé sans codage et sans balises HTML.

```
https://www.saferpay.com/vt/Pay.asp?  
DATA=%3cIDP%20ACCOUNTID%3d%2299867%2d94913159%22%20ALLOWCOLLECT%3d%22no%22%  
20AMOUNT%3d%22100%22%20BACKLINK%3d%22%2e%22%20CURRENCY%3d%22DEM%22%20DELIVE  
RY%3d%22no%22%20DESCRIPTION%3d%22Testkauf%20Warenkorb%22%20EXPIRATION%3d%22  
20010408%2012%3a13%3a50%22%20FAILLINK%3d%22%2e%22%20KEYID%3d%220%2d37217%2d  
ea645c3f3f0911d583d70050da413f31%22%20MSGTYPE%3d%22PayInit%22%20SUCCESSLINK  
%3d%22%2e%22%20TOKEN%3d%22ea645c5d3f0911d583d70050da413f31%22%2f%3e&SIGNATU  
RE=2f1ec1fa51002817941c22e98b9047422ba9ff8fce8b61dab8208a5aa8c82be7cda02ff8  
a66930481fc19b16d05e7bcedd2b0e5be98fecad3d48bd43916a502f
```

Si une erreur apparaît, le marqueur « ERROR » est transmis accompagné d'une description de l'erreur :

```
ERROR: Missing AMOUNT attribute
```

4.4 Vérification de la réponse d'autorisation via l'adresse VerifyPayConfirm

Le SUCCESSLINK est exécuté une fois le paiement réussi. Les paramètres DATA et SIGNATURE sont transmis en même temps que lui via GET au système de la boutique en ligne. Le paramètre DATA contient le message PayConfirm qui comporte les détails de la réponse d'autorisation et le paramètre SIGNATURE contient la clé avec laquelle DATA de Saferpay a été signé. Afin d'exclure toute manipulation de la réponse d'autorisation, les deux paramètres doivent être envoyés dès réception à l'adresse VerifyPayConfirm de la Saferpay HI, qui répond ensuite à la requête par OK ou ERROR.

4.5 Exemple de VerifyPayConfirm

Les paramètres DATA et SIGNATURE sont envoyés via GET ou POST au Saferpay Gateway:

Exemple d'exécution (GET):

```
https://www.saferpay.com/hosting/VerifyPayConfirm.asp? DATA=...
```

Si la signature numérique correspond totalement aux valeurs contenues dans DATA, la vérification réussie est indiquée par OK et comprend l'ID et le TOKEN Saferpay :

```
OK: ID=56a77rg243asfhmkq3r&TOKEN=%3e235462FA23C4FE4AF65...
```

En cas d'erreur, le texte « ERROR » est renvoyé accompagné d'une description de l'erreur :

```
ERROR: Possible manipulation
```

4.6 Comptabilisation d'un paiement via l'adresse PayComplete

La comptabilisation d'une autorisation s'effectue via l'exécution de l'adresse CreatePayComplete de la HI. Pour cela, les paramètres ACCOUNTID et ID doivent être envoyés via GET ou POST à la Saferpay HI. Après réception du message PayComplete, la HI répond par OK ou ERROR.

Une comptabilisation avec un montant inférieur ou l'annulation d'un paiement via l'adresse CreatePayComplete de la HI n'est pas possible pour des raisons de sécurité. C'est la raison pour laquelle les paramètres ACTION et AMOUNT sont ignorés.

Exemple d'exécution (GET):

```
https://www.saferpay.com/hosting/PayCompleteV2.asp?  
ACCOUNTID=401860-17795278&ID=5sfhmkq3rg54345abcd234&spPassword=XAjc3Kna*
```

Si l'opération a pu être effectuée avec succès, OK est envoyé :

```
OK
```

En cas d'erreur, le texte ERROR est renvoyé accompagné d'une description de l'erreur :

```
ERROR: Error description
```

** La transmission du paramètre spPassword n'est nécessaire que pour le compte -test. Pour les comptes en exploitation, ce paramètre n'est pas nécessaire !*

4.7 Exécution de la Payment Page via l'adresse Redirect

Il est parfois impossible de recevoir les données contenues dans la réponse de la Saferpayhttps Interface, car les paramètres du serveur Internet ne l'autorisent pas ou peut-être parce que le système de boutique en ligne utilisé ne fonctionne qu'avec JavaScript. Dans une telle situation, l'adresse Redirect de la HI peut être utilisée pour exécuter la PP.

4.7.1 Inconvénients de Redirect

Comme les données de paiement doivent être redirigées pour cette procédure vers le site Internet, il existe théoriquement la possibilité qu'un utilisateur averti d'Internet puisse les manipuler. De la même manière, le message PayConfirm pourrait être modifié ou simulé.

Utilisez l'adresse Redirect uniquement en dernier recours. L'adresse CreatePayInIt doit être si possible toujours utilisée !

4.7.2 Adresse https Interface

La Saferpayhttps Interface est accessible via cette adresse Internet :

Exécution de la Payment Page via Redirect:

<https://www.saferpay.com/hosting/Redirect.asp>

4.7.3 Déroulement de la transaction avec Redirect

Les paramètres PayInIt nécessaires au paiement sont transmis via GET ou POST à la Saferpay HI. En conséquence, un URL de paiement est généré sur la HI et la Payment Page est exécutée directement via « redirect ». Après que le client a effectué avec succès son paiement via la PP, la réponse d'autorisation (message PayConfirm) est tout d'abord envoyée à la Saferpay HI. Là, elle est soumise au contrôle de plausibilité/manipulation (VerifyPayConfirm) et le résultat de la vérification est ensuite transmis via GET au système de la boutique en ligne avec le SUCCESSLINK. Le résultat de la vérification est indiqué au système de la boutique en ligne avec le paramètre RESULT. RESULT peut contenir les valeurs « 0 » en cas de réussite ou « 1 » en cas de manipulation ou d'erreur.

Exemple d'exécution (GET):

<https://www.saferpay.com/hosting/Redirect.asp? AMOUNT=1095&...>

Grâce à l'exécution, le client est basculé directement sur son navigateur vers la PP afin d'effectuer son paiement.

Suite au paiement réussi, la vérification de la signature numérique a lieu sur la HI. Ensuite, le résultat est transmis via GET au SUCCESSLINK. Si la vérification est réussie, le résultat affiché est RESULT=0 et d'autres paramètres PayConfirm sont disponibles avec les détails de l'autorisation.

http://www.shop.de/kasse_ok.php? RESULT=0&ID=J89HBV...

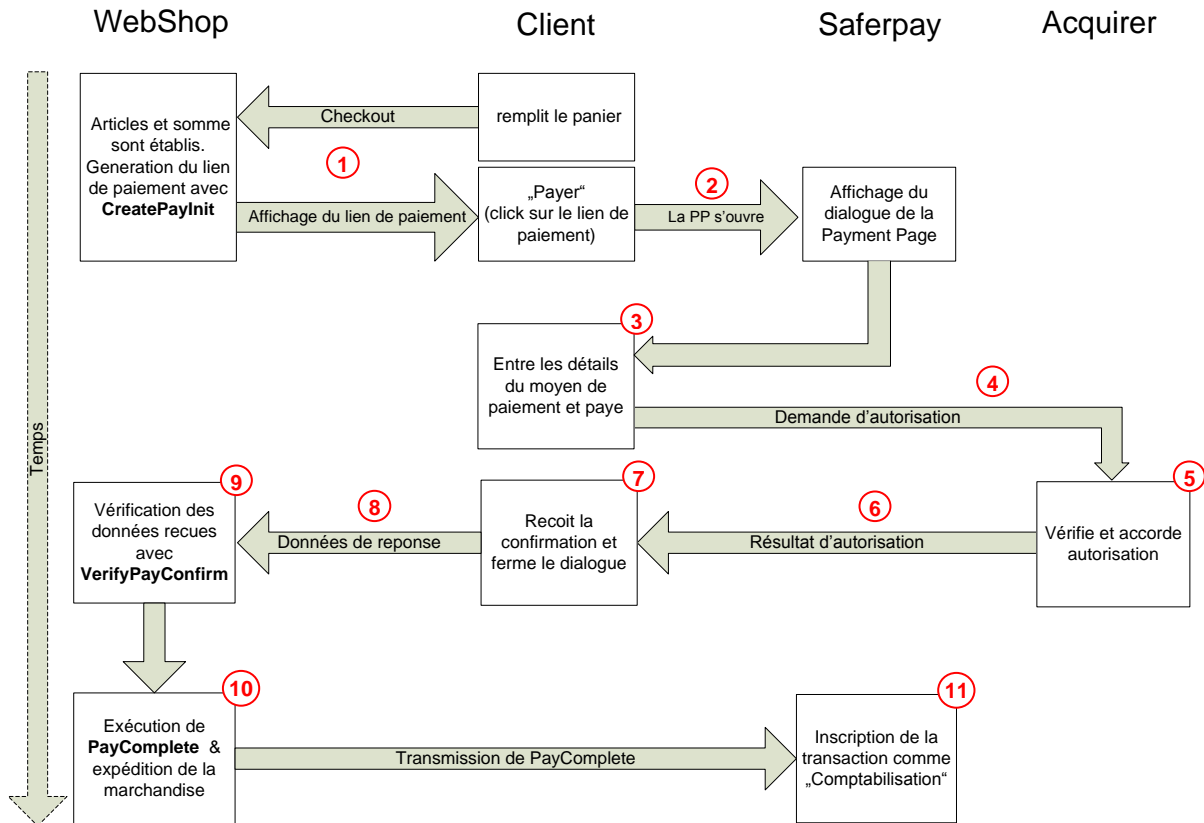
Si la vérification de la signature numérique échoue, le résultat transmis est uniquement RESULT=1 :

http://www.shop.de/kasse_ok.pl? Session=123&RESULT=1

5 Etapes du traitement

5.1 Résumé

Le schéma ci-dessous montre la procédure pour un paiement en ligne réussi avec la SaferpayPayment Page :



5.2 Description de la procédure

- ① Dès que le client a rempli son panier et que le montant à payer est fixé, la boutique en ligne génère un lien de paiement (CreatPayInit). La boutique en ligne représente le lien de paiement sur la confirmation de commande sous forme d'un bouton « Payer » ou similaire.
- ② Le client clique sur « Payer » et la PP s'ouvre.
- ③ Le formulaire de la PP propose au client un moyen de paiement. Le client choisit lui-même parmi les possibilités son moyen de paiement pour l'opération et saisit les informations requises figurant sur son moyen de paiement.
- ④ L'autorisation en ligne du moyen de paiement est ensuite effectuée.
- ⑤ La société chargée du traitement des cartes vérifie la demande d'autorisation et délivre l'autorisation.
- ⑥ La société chargée du traitement des cartes transmet la réponse d'autorisation.
- ⑦ Sur la PP, une confirmation du paiement est affichée et le client est invité à terminer la procédure en cliquant sur « Terminer ».

- 8 La PP se ferme et le client est rebasculé via le SUCCESSLINK vers la boutique en ligne qui reçoit les données de la réponse d'autorisation.

Dans de rares cas, il est possible que SUCCESSLINK soit appelé plusieurs fois. Les raisons peuvent être de différente nature et cela ne peut pas toujours être empêché par le biais de dispositifs techniques. Pour reconnaître de tels cas, il est recommandé d'utiliser NOTIFYURL ou de vérifier l'univocité des valeurs de retour, par exemple à l'aide d'un Shop Session ID lié au SUCCESSLINK via le GET. Vous pouvez ainsi vous assurer que le shop n'a enregistré aucune commande en double et que les actions de suivie sont correctes, par exemple que PayComplete n'est appelé qu'une seule fois. .

- 9 Le système du commerçant vérifie la confirmation de paiement (VerifyPayConfirm) et l'enregistre avec les informations liées à l'opération.

- 10 La somme est débitée (PayComplete) et la commande peut être effectuée.

La comptabilisation d'une réservation est absolument obligatoire pour le bouclage journalier. Ce dernier ne prend en compte que les transactions dont le statut est « Comptabilisation » et les transmet pour paiement à l'acquirer. L'argent est crédité sous la forme d'une ligne commune sur le compte d'entreprise du commerçant. Le commerçant reçoit de l'établissement de cartes de crédit une liste des décomptes.

Selon le type d'opération, la comptabilisation peut être effectuée ultérieurement, généralement dans les six jours ce qui correspond à la durée moyenne de validité d'une réservation. Comme cela varie selon les acquirers et les moyens de paiement, veuillez vous renseigner directement auprès de votre acquirer pour connaître vos conditions.

Le bouclage journalier peut être déclenché manuellement ou automatiquement.

- 11 Le PayComplete fait passer le statut de la transaction de « Réservation » à « Comptabilisation ». Le statut de la transaction est affiché dans le Saferpay Backoffice Journal.

6 Paramètres

6.1 Paramètres Paylnit

Le tableau répertorie les paramètres disponibles pour le message CreatePaylnit. Sauf mention contraire, l'utilisation de chaque paramètre est obligatoire.

Paramètre	Format	Description
ACCOUNTID	ns[..15]	Le numéro de compte Saferpay du commerçant pour cette transaction. Par exemple « 401860-17795278 » pour le compte -test Saferpay.
AMOUNT	n[..8]	Somme à payer dans la plus petite unité de la devise. Par exemple : « 1230 » correspond à 12,30 en euro.
CURRENCY	a [3]	Code à trois caractères de la devise selon ISO 4217. Par exemple : « CHF » ou « EUR »
DESCRIPTION	ans[..50]	Contient une description de l'offre affichée sur la PP et aussi dans les détails de la transaction dans le Saferpay Backoffice une fois le paiement réussi. Pour les clients qui utilisent MasterPass, elle apparaît aussi dans le Shopping Cart local. Un formatage de l'affichage de la PP via les caractères de commande n'est pas possible. L'utilisation de plus de 50 signes est ainsi déconseillée même si techniquement possible. La valeur transmise doit obligatoirement être codée en HTML pour un traitement correct des caractères spéciaux.
ORDERID	ans[..80]	<i>Optionnel, Paramètre obligatoire avec le moyen de paiement giropay</i> ORDERID contient le numéro de référence d'un paiement. L'ORDERID doit être sans équivoque afin de permettre un référencement ultérieur. Saferpay peut afficher 80 signes pour l'ORDERID ce qui n'est généralement pas possible chez les acquiers. Les chaînes de signes trop longues y sont sectionnées. 12 signes ont dans la pratique fait leurs preuves. En cas de doute, veuillez vous informer auprès de votre acquier sur le nombre de signes affichés. Lors de l'utilisation du service Alias de PostFinance, une valeur unique est obligatoire.
VTCONFIG	an[..20]	<i>Optionnel</i> VTCONFIG détermine la configuration à utiliser pour la PP. Plusieurs configurations pour la PP peuvent être créées dans le Saferpay Backoffice. Quelques options de la PP, comme l'ajout d'un logo, ne peuvent être activées que là-bas. Valeur : Nom de la configuration dans le Saferpay Backoffice.
SUCCESSLINK	ans[..1024]	URL vers lequel le client est renvoyé suite à la réservation réussie via le browser Redirect. Le message de confirmation (PayConfirm) est attaché par Saferpay à cette adresse via GET.
FAILLINK	ans[..1024]	URL vers lequel le client est renvoyé via le browserRedirect lorsque l'autorisation n'a pas pu être effectuée.
BACKLINK	ans[..1024]	<i>Optionnel</i> URL vers lequel le client est renvoyé via le browserRedirect lorsque l'autorisation est interrompue.

Paramètre	Format	Description
NOTIFYURL	ans[..1024]	<p><i>Optionnel</i></p> <p>URL pleinement qualifié qui est directement exécuté par les serveurs Saferpay suite à une autorisation réussie et qui transmet le message de confirmation (PayConfirm) via POST. L'exécution n'est possible que sur les ports standards pour http (80) ou https (443). Les autres ports ne fonctionnent pas. En règle générale, l'utilisation de NOTIFYURL est conseillée afin de s'assurer une réponse du système du commerçant indépendamment des erreurs ou problèmes éventuels du côté du client. Afin de faciliter le référencement d'une commande, la transmission du Session ID d'une boutique en ligne en tant que paramètre GET au NOTIFYURL a fait ses preuves.</p> <div style="border: 1px solid red; padding: 5px; margin-top: 10px;"> <p>*ATTENTION! L'utilisation de NOTIFYURL est recommandée en toutes circonstances afin d'éviter des erreurs lors du démarrage du SUCCESSLINK par le client. Nous vous prions de veiller à ce que votre système reçoive la confirmation de paiement (PayConfirm) deux fois également dans le cas normal. Le paramètre ORDERID peut ici être utile pour l'identification. En outre, l'utilisation de NOTIFYURL est absolument indispensable pour le moyen de paiement Masterpass (voir chapitre 6.1.4).</p> </div>
AUTOCLOSE	n[..2]	<p><i>Optionnel</i></p> <p>Ferme automatiquement la PP après autorisation réussie et exécute le SUCCESSLINK. La valeur transmise indique le temps en secondes qui est affiché sur la page de compte à rebours avant fermeture.</p> <p>Valeur conseillée : « 0 »</p>
CCNAME	a[..3]	<p><i>Optionnel</i></p> <p>Active le champ pour la saisie du nom du titulaire de la carte sur la PP.</p> <p>Valeurs : « yes » ou « no »</p>
NOTIFYADDRESS	ans[..50]	<p><i>Optionnel</i></p> <p>Adresse électronique du commerçant. Saferpay envoie un courriel de confirmation à cette adresse après une réservation réussie.</p>
USERNOTIFY	ans[..50]	<p><i>Optionnel</i></p> <p>Adresse électronique du client. Saferpay envoie un courriel de confirmation à cette adresse après une réservation réussie.</p>
LANGID	a[2]	<p><i>Optionnel</i></p> <p>Code de la langue selon la norme ISO 639-1. Indique la langue de la PP. Une liste des codes des langues disponibles se trouve ci-dessous au chapitre 6.1.1.</p> <p>Sans LANGID, la langue du navigateur du client est affichée sur la PP. Si elle n'est pas reconnue, la langue par défaut est l'allemand.</p>

Paramètre	Format	Description
SHOWLANGUAGES	A[..3]	<i>Optionnel</i> Active le menu pour le choix de la langue sur la PP. Valeurs : « yes » ou « no »
PAYMENTMETHODS	ns[..100]	<i>Optionnel</i> Indique les moyens de paiement affichés sur la PP. Par défaut, tous les moyens de paiement activés sur un terminal sont affichés sur la PP. PAYMENTMETHODS transmet un ou plusieurs ID de moyens de paiement séparés par une virgule. Par exemple : « 1,2,13 » active MasterCard, Visa et ELV Une liste de tous les ID des moyens de paiement se trouve plus bas dans le chapitre 6.1.2.
DURATION	n[14]	<i>Optionnel</i> Limite la durée de validité du lien de paiement. Après expiration, le lien perd sa validité. Format: AAAAMMJJhhmmss
PREAUTH	a[..3]	<i>Optionnel</i> Est nécessaire pour signaler une transaction comme préautorisation. Les transactions préautorisées peuvent être capturées jusqu'à 30 jours après l'autorisation par PayComplete. Valeurs: „yes“ ou „no“ Si le paramètre n'est pas transmis, alors une autorisation finale est effectuée.
<p>*ATTENTION! La préautorisation est disponible uniquement comme moyen de paiement MasterCard ainsi que pour les techniciens de SIX et de ConCardis!</p>		
WALLETS	ns[..100]	<i>Optionnel</i> Permet la redirection immédiate du client vers la page d'accès de son wallet, où il pourra sélectionner le moyen de paiement qui fera ensuite l'objet d'une autorisation automatique. Si PAYMENTMETHODS apparaît aussi simultanément, il est traité avec le même niveau de priorité. Valeur: «MASTERPASS»

CARDREFID

ans[.40]

Optionnel

Valeur de remplacement pour le numéro de carte de crédit et la date d'expiration ou la relation bancaire (uniquement pour l'ELV allemand). La valeur de remplacement peut être attribuée par l'application de commerce en ligne ou par Saferpay (new). Utilisation possible uniquement avec le service « Saferpay Secure Card Data ».

Valeurs : Valeur de remplacement sans équivoque ou « new »

Paramètre	Format	Description
DELIVERY	a[..3]	<i>Optionnel</i> Indique si un formulaire d'adresse est affiché. Valeurs : « yes » (default), « no »
APPEARANCE	an[..7]	<i>Optionnel</i> Adapte l'affichage de la PP à l'écran utilisé. Valeurs : « auto » (default), « mobile », « desktop », « embedded »
CSSURL	ans[..1024]	<i>Optionnel, peut uniquement être utilisé avec APPEARANCE «mobile» ou «embedded»</i> Contient l'URL https totalement qualifiée vers le fichier CSS pour l'adaptation de la PP en vue mobile, voir le chapitre 7 .
MANDATEID	an[..35]	<i>Optionnel</i> Référence de mandat pour l'affichage dans le texte du mandat pour les paiements avec ELV. La valeur doit être unique. Par défaut l'ID de transaction de Saferpay est utilisé.
RECURRING	a[..3]	<i>Optionnel</i> Identifie la demande comme autorisation initiale d'un paiement récurrent. Lors de paiements ELV, occasionne l'affichage d'un texte de mandat spécifique pour les paiements récurrents. Valeurs: "yes" ou "no" (défaut)
Affichage des Conditions générales**:		
TERMSCHECKBOXACTIVE	a[..8]	<i>Optionnel, doit être accompagné de TERMSURL</i> Contient une case à cocher pour accepter les Conditions générales du négociant. Valeurs: «yes» ou «no» (défaut)
TERMSURL	ans[..1024]	<i>Optionnel, doit être accompagné de TERMSCHECKBOXACTIVE</i> Contient l'URL menant aux Conditions générales du négociant.
Options d'adresse:		
ADDRESS	a[..8]	<i>Optionnel</i> Indique, si un formulaire d'adresse est pour une adresse fournisseur, client ou de facturation. Valeurs : « DELIVERY », « CUSTOMER », « BILLING »
Si l'adresse est déjà disponible, elle peut être transmise au formulaire d'adresse. Les caractères spéciaux contenus dans les valeurs des paramètres suivants doivent être encodés en HTML Entity:		
COMPANY	ans[..50]	<i>Optionnel</i> Nom / Dénomination de l'entreprise
GENDER	a[1]	<i>Optionnel</i> Sexe Valeurs : « f », « m », « c » (company)
FIRSTNAME	ans[..50]	<i>Optionnel</i> Prénom

Paramètre	Format	Description
LASTNAME	ans[..50]	<i>Optionnel</i> Nom
STREET	ans[..50]	<i>Optionnel</i> Rue
ZIP	an[..10]	<i>Optionnel</i> Code postal
CITY	ans[..50]	<i>Optionnel</i> Lieu
COUNTRY	a[2]	<i>Optionnel</i> Code pays selon la norme ISO 3166.
STATE	a[2]	<i>Optionnel</i> Indication d'un Etat fédéral ou d'une province. Correspond aux deux derniers signes selon la norme ISO 3166-2.
EMAIL	ans[..50]	<i>Optionnel</i> Adresse électronique
PHONE	ns[..20]	<i>Optionnel</i> Numéro de téléphone

* Pour l'utilisation de CARDREFID="new", une valeur initiale numérique pour le compte doit d'abord être enregistrée par Saferpay. Contactez à cet effet integration.saferpay@six-payment-services.com.

** Pour les transactions MasterPass, la mention d'acceptation des Conditions générales ne s'affiche pas. Les paramètres des Conditions générales sont donc ignorés lors de la sélection de MasterPass.

6.1.1 Codes langues pour LANGID

Code	Nom	Langue
de	Deutsch	Allemand
en	English	Anglais
fr	Français	Français
da	Dansk	Danois
cs	Čeština	Tchèque
es	Español	Espagnol
hr	Hrvatski	Croate
it	Italiano	Italien
hu	Magyar	Hongrois
nl	Nederlands	Néerlandais
no	Norsk	Norvégien
pl	Polski	Polonais
pt	Português	Portugais
ru	Русский	Russe
ro	Română	Romain
sk	Slovenský	Slovaque
sl	slovenščina	Slovène
fi	Suomi	Finnois
sv	Svenska	Suédois
tr	Türkçe	Turque
el	Ελληνικές	Grec
ja	日本語	Japonais
zh	中国語	Chinois

6.1.2 ID des moyens de paiement pour - PAYMENTMETHODS

ID des moyens de paiement	Moyen de paiement
1	MasterCard
2	Visa
3	American Express
4	Diners Club
5	JCB
6	Saferpay Test Card
8	Bonus Card
9	PostFinance E-Finance
10	PostFinance Card
11	Maestro International
12	MyOne
13	Prélèvement
14	Facture
15	Virement express
16	PayPal
17	giropay
18	iDEAL
20	Homebanking AT (eps)
22	ePrzelewy

6.1.3 Paramètres PayConfirm de la réponse d'autorisation

Avec la réponse d'autorisation, le message PayConfirm est transmis. Celui-ci peut contenir les paramètres suivants:

Paramètre	Format	Description
MSGTYPE	a[..30]	Contient toujours la valeur "PayConfirm".
VTVERIFY	ans[..40]	Peut contenir des informations supplémentaires sur le traitement de la transaction. Valeur par défaut : "(obsolete)"
KEYID	ans[..40]	Identifiant de la clé avec laquelle la signature a été générée.
ID	an[28]	Identifiant Saferpay de transaction sans équivoque.
TOKEN	ans[..40]	Peut contenir des informations supplémentaires sur le traitement de la transaction. Valeur par défaut : "(unused)"
ACCOUNTID	ns[..15]	Le numéro de compte Saferpay du commerçant pour cette transaction. Par exemple « 401860-17795278 » pour le compte -test Saferpay.
AMOUNT	n[..8]	Somme à payer dans la plus petite unité de la devise. Par exemple : « 1230 » correspond à 12,30 en euro.
CURRENCY	a[3]	Code à trois caractères de la devise selon ISO 4217. Par exemple : « CHF » ou « EUR »
DCCAMOUNT	n[..8]	<i>Optionnel (uniquement pour les paiements avec Dynamic Currency Conversion)</i> Montant de la monnaie de la carte de crédit dans la plus petite unité monétaire, par exemple : « 1230 » correspond à 12,30 EUR.
DCCCURRENCY	a[3]	<i>Optionnel (uniquement pour les paiements avec Dynamic Currency Conversion)</i> Code ISO 4217 à trois lettres de la monnaie de carte de crédit. Par exemple : « CHF » ou « EUR »
CARDREFID	ans[..40]	<i>Optionnel</i> (uniquement lorsque le paramètre est également transmis lors de l'exécution de CreatePayInit.) Contient la valeur de remplacement pour le numéro de carte de crédit et la date d'expiration ou la relation bancaire (uniquement pour l'ELV allemand),
SCDRESULT	n[..4]	<i>Optionnel (uniquement lorsque le paramètre CARDREFID est indiqué lors de l'exécution de CreatePayInit)</i> Contient le code réponse de l'enregistrement dans le SCD : 0 Enregistrement réussi. 7000 Erreur générale (voir DESCRIPTION). 7001 Demande n'a pas pu être entièrement traitée. 7002 Type de carte non disponible sur le terminal. 7003 Paramètre avec contenu ou format invalides. 7004 CARDREFID introuvable (uniquement pour l'autorisation). 7005 Paramètre manquant dans la demande. 7006 CARDREFID déjà existant. 7007 Aucun droit disponible pour le SCD.
PROVIDERID	n[..4]	Contient le Provider ID de la société chargée du traitement du moyen de paiement.
PROVIDERNAME	ans[..40]	Contient le nom de la société chargée du traitement du moyen de paiement.
PAYMENTMETHOD	n[2]	Contient le ID du moyen de paiement.
ORDERID	an[..39]	<i>Optionnel</i> Numéro de référence du commerçant transmis lors de CreatePayInit.

Paramètre	Format	Description
IP	ns[..15]	<i>Optionnel*</i> Contient l'adresse IP du client. Disponible uniquement pour un SaferpayRisk Management existant.
IPCOUNTRY	a[2]	<i>Optionnel*</i> Pays d'origine de l'adresse IP du payeur d'après la norme ISO 3166. Code pays (Ex.: CH, DE, AT). Si une classification n'est pas possible, la valeur est "IX". Exemple : "DE"
CCCOUNTRY	a[2]	<i>Optionnel*</i> Pays d'origine de la carte d'après la norme ISO 3166. Si une classification n'est pas possible, CCCOUNTRY n'est pas inclus dans la réponse. Exemple : "DE"
MPI_LIABILITYSHIFT	a[..3]	<i>Optionnel**</i> Indique, s'il existe un transfert de responsabilité d'un point de vue purement technique. Valeurs : "yes" ou "no" <div style="border: 1px solid red; padding: 5px; margin-top: 10px;">Attention !Tous les acquirers ne sont pas en mesure de vérifier le transfert de responsabilité pendant l'autorisation et excluent celle-ci le cas échéant directement dans la réponse d'autorisation. Voilà pourquoi, il est possible que pour des raisons contractuelles le transfert de responsabilité ne soit pas garanti par l'acquirer, même si MPI_LIABILITYSHIFT et ECI affirment le contraire. Demandez, si besoin est, directement à votre acquirer, s'il en est capable.</div>
ECI	n[1]	<i>Optionnel**</i> Electronic Commerce Indicator Nécessaire pour l'identification des transactions avec 3-D Secure (« Verified by Visa », « MasterCard SecureCode »): 0 = Paiement sur Internet sans transfert de responsabilité 1 = Paiement avec 3-D Secure avec autorisation 2 = Paiement avec 3-D Secure, pas de participation à la procédure
XID	ans[28]	<i>Optionnel**</i> Cette séquence de caractères en Base 64 est attribuée par le MPI et renvoie à la procédure avec le protocole 3-D Secure.
CAVV	ans[28]	<i>Optionnel**</i> CardholderAuthenticationVerification Value Contient ici la valeur UCAF pour une MasterCard, le code AEVV pour une American Express. Saferpay utilise indépendamment du type de carte de crédit la valeur CAVV.
IBAN	an[22]	<i>Optionnel</i> Contient l'IBAN de la demande. Par exemple: "DE77970000010123456789"

Paramètre	Format	Description
MANDATEID	ans[..35]	<i>Optionnel</i> Contient la référence de mandat du paiement ELV.
CREDITORID	ans[..35]	<i>Optionnel</i> Contient l'identifiant créancier affiché dans le texte du mandat lors des paiements ELV.

* Uniquement disponible pour un SaferpayRisk Management existant

** Paramètre 3-D Secure, la condition préalable est la participation à la procédure 3-D Secure (« Verified by Visa », « MasterCard SecureCode », « American Express SafeKey »)

Important! Pour se protéger des risques de manipulation, il est **vivement conseillé** de vérifier en plus de la signature également les paramètres **ACCOUNTID**, **ORDERID**, **AMOUNT** et **CURRENCY** transmis avec le message PayConfirm. Ceux-ci doivent avoir les mêmes valeurs que celles qui ont été attribuées à l'origine au message Paylnit !

6.1.4 Paramètres d'adresse lors de l'utilisation de Masterpass

Si vous utilisez le Masterpass Wallet, alors les données d'adresse du client seront transmises via NOTIFYURL avec le reste des données.

Paramètre	Format	Description
GENDER	a[1]	Sexe Valeur: "f", "m"
FIRSTNAME	ans[..50]	Prénom
LASTNAME	ans[..50]	Nom
DATEOFBIRTH	n[8]	Date de naissance du titulaire de la carte
STREET	ans[..50]	Rue
ZIP	an[..10]	Code postal
CITY	ans[..50]	Lieu
COUNTRY	a[2]	Code pays d'après la norme ISO 3166.
STATE	a[2]	Code d'un Etat fédéral ou d'une province. Correspond aux deux derniers signes d'après la norme ISO 3166-2.
EMAIL	ans[..50]	Adresse e-mail
PHONE	ns[..20]	Numéro de téléphone
ADDRESSADDITION	an[..50]	Paramètre d'adresse supplémentaire
DELIVERY_GENDER	a[1]	Sexe pour l'adresse de livraison
DELIVERY_FIRSTNAME	ans[..50]	Prénom pour l'adresse de livraison
DELIVERY_LASTNAME	ans[..50]	Nom pour l'adresse de livraison
DELIVERY_STREET	ans[..50]	Rue de l'adresse de livraison
DELIVERY_ADDRESSADDITION	an[..50]	Supplément pour l'adresse de livraison
DELIVERY_ZIP	an[..10]	Code postal de l'adresse de livraison
DELIVERY_CITY	ans[..50]	Lieu de l'adresse de livraison
DELIVERY_COUNTRY	a[2]	Code pays d'après la norme ISO 3166 pour l'adresse de livraison
DELIVERY_PHONE	ns[..50]	Numéro de téléphone pour l'adresse de livraison

6.2 Demande VerifyPayConfirm

La demande VerifyPayConfirm pour vérification de la signature de la réponse d'autorisation contient les paramètres suivants :

Paramètre	Format	Description
DATA	ans[..1024]	Message XML reçu avec le message PayConfirm, inchangé.
SIGNATURE	an[28]	Contient la signature pour DATA
ACCOUNTID	ns[..15]	<i>En option avec l'utilisation d'une LIB</i> Le numéro de compte Saferpay du commerçant pour cette transaction. Par exemple "401860-17795278" pour le compte-test Saferpay.

6.3 Demande PayComplete

Les paramètres suivants sont disponibles pour un message CreatePayComplete. Sauf mention contraire, chaque paramètre est à utiliser obligatoirement :

Paramètre	Format	Description
ID	an[28]	Identifiant de transaction Saferpay du message PayConfirm. Paramètre obligatoire sauf si ACTION=CloseBatch.
AMOUNT	n[..8]	<i>Optionnel</i> Somme à payer dans la plus petite unité de la devise. Par exemple : « 1230 » correspond à 12,30 en euro.
ACCOUNTID	ns[..15]	Le numéro de compte Saferpay du commerçant pour cette transaction. Par exemple « 401860-17795278 » pour le compte-test Saferpay.
ACTION		<i>Optionnel</i> Indique une possibilité de traitement élargie. Valeurs possibles : « Settlement », « CloseBatch », « Cancel »

Settlement

Indique au système Saferpay qu'il faut passer le statut de la transaction de Réservation à Comptabilisation. La transaction est ainsi transmise lors du bouclage journalier suivant à l'acquiescer pour son paiement. Grâce au paramètre AMOUNT, il est possible d'indiquer lors de la comptabilisation un montant inférieur que lors de la réservation. La différence est ainsi automatiquement annulée. Il n'est pas possible de débiter plus que le montant autorisé.

CloseBatch

Indique à Saferpay d'exécuter le bouclage journalier pour l'ACCOUNTID donné. Si le paramètre ID est transmis, l'exécution échoue.

Cancel

Avec cette commande, il est possible de rejeter une réservation ou d'annuler une comptabilisation tant qu'elle n'a pas été traitée avec le bouclage journalier. S'il s'agit d'une réservation, la transaction est encore affichée 6 jours après son rejet dans le Saferpay Backoffice sous la rubrique « Réservations rejetées ». Après ce délai, elle est effacée de la base de données. Les comptabilisations annulées sont par contre conservées, elles sont marquées comme annulées et sont visibles dans le Backoffice.

Sans paramétrage d'ACTION, la valeur standard est ACTION="Settlement".

6.4 Réponse PayComplete

La réponse à la demande de comptabilisation contient les paramètres suivants :

Paramètre	Format	Description
MSGTYPE	a[..30]	Contient toujours la valeur "PayConfirm".
ID	an[28]	Identifiant de transaction Saferpay
RESULT	n[..4]	Contient le résultat de la demande d'autorisation. 0 = Demande traitée avec succès. ≠0 = Demande traitée sans succès.
MESSAGE	ans[..30]	Contient une réponse texte à la demande de comptabilisation.
AUTHMESSAGE	ans[..30]	Contient une réponse texte concernant l'autorisation.

7 Adaptation avec feuilles de styles en cascade

Il est possible de modifier totalement la présentation de la Payment Page en vue «mobile» à l'aide d'un fichier CSS. Ce fichier CSS n'a pas d'influence sur la présentation de la Payment Page en vue Desktop. Celle-ci peut être modifiée à l'aide du paramètre **VTCONFIG** et d'un modèle adéquat dans la rubrique «Configuration de la Payment Page» dans le Backoffice de Saferpay.

7.1 Variantes de style

Deux variantes sont disponibles pour adapter la Payment Page avec une feuille de style.

7.1.1 Modifier la Payment Page mobile de Saferpay

La première variante recourt au paramètre PayInit APPEARANCE=„mobile“ et transmet le chemin vers le fichier CSS au paramètre PayInit CSSURL.

Exemple:

```
APPEARANCE="mobile"  
CSSURL="https://link.to.mydesign.css"
```

Ensuite, la Payment Page mobile est chargée avec la présentation standard de Saferpay et les présentations prescrites par Saferpay sont écrasées par les données du fichier CSS.

7.1.2 Présentation totalement nouvelle pour la Payment Page

Alternative à la première variante: le paramètre PayInit APPEARANCE=„embedded“ est utilisé avec le paramètre PayInit CSSURL.

Exemple:

```
APPEARANCE="embedded"  
CSSURL="https://link.to.mydesign.css"
```

La Payment Page mobile se présente ensuite dans la vue HTML par défaut du navigateur et peut être totalement modifiée à l'aide du fichier CSS. Les éléments HTML sont normalisés, si bien qu'ils s'affichent de manière uniforme et conforme aux standards modernes.

7.2 Taille de l'Inline Frame

La largeur et la hauteur de l'iFrame sont transmises à la page Internet de la boutique via la fonction HTML5 `postMessage`.

Exemple JSON:

```
{
  "message": "css",
  "height": 450,
  "width": 650
}
```

Exemple de réception du message (jQuery):

```
$(window).bind("message", function (e) {
    $("#iframe").css("height", e.originalEvent.data.height + "px");
});
```

Etant donné que toutes les pages Internet ne fournissent pas d'informations sur leur taille, il est recommandé de définir une hauteur et une largeur minimale sur la page de la boutique.

7.3 Utilisation du CSS

Il convient de tenir compte des informations suivantes lors de l'adaptation de la Payment Page à l'aide de CSS.

7.3.1 Element Name

Le nom de l'élément peut être utilisé conformément à la spécification CSS.

Exemple:

```
h1{
  text-decoration: underline;
}
```

7.3.2 Class Name

Il convient d'utiliser les noms de catégories CSS définies par Saferpay figurant au [chapitre 7.5](#).

Exemple:

```
.form-group
{
font-family: "Times New Roman", serif;
font-style: italic;
}
```

7.3.3 Element ID

L'ID de l'élément ne doit pas être utilisé, car les ID peuvent être modifiées sans préavis.

7.3.4 Element Attribute

Les attributs d'éléments (name, value, data-*, etc.) ne doivent pas être utilisés, car ils peuvent être modifiés sans préavis.

7.3.5 Sélecteurs CSS

Le système supporte tous les sélecteurs CSS pour CSS1, CSS2 et CSS3.

7.4 Informations importantes pour l'utilisation de CSS

- La feuille de style à laquelle renvoie le paramètre Paylnit **CSSURL** doit être enregistrée sur un serveur Web supportant https.
- Au sein de cette feuille de style, il faut veiller à ce que les graphiques soient chargés via «https://»,. faute de quoi le navigateur affichera un message d'erreur. Par exemple: „[...] *this page includes other resources which are not secure.* [...]».
- Si le navigateur du Shopper bloque les «third party cookies», la Payment Page de Saferpay affiche la page d'erreur «*Cookies are disabled*»..
- Pendant le chargement de l'iFame, il est recommandé d'afficher la barre de progression.
-
- En cas de retour à la page Success, Abort ou Fail, il est recommandé de sortir du Frame.

7.5 Noms de catégories CSS

7.5.1 Catégories générales

Html

- saferpay-paymentpage

Header

- img-logo

Footer

- btn-back
- btn-abort
- btn-next

- navitem
- navitem-back
- navitem-abort
- navitem-next

Boîtes d'affichage

- box-content
 - box-shop
 - box-information
 - box-success
 - box-error
-
- box-information-required-fields
 - img-shop
 - text-information
 - text-success
 - text-error
 - icon-information
 - icon-success
 - icon-error

Formulaire

- form-group
- form-label
- form-input
- form-col-small
- form-col-large
- input-required
- input-large
- input-medium
- input-small
- icon-required

Formulaire validation

- validation-summary-errors
- label-validation-error
- input-validation-error

Catégories de base

- box
- btn
- img
- icon
- text

7.5.2 Spécifique pour Payment Selection

- page-paymentsselection
- paymentgroup
 - paymentgroup-creditcard
 - paymentgroup-onlinebanking
 - paymentgroup-card
 - paymentgroup-onlinepaymentservice
 - paymentgroup-directdebit
 - paymentgroup-invoice
 - paymentgroup-wallet
- btn-select
- btn-creditcard-visa
- btn-creditcard-mastercard
- btn-creditcard-maestro
- btn-creditcard-amex
- btn-creditcard-jcb
- btn-creditcard-dinersclub
- btn-creditcard-saferpay
- btn-onlinebanking-sofort
- btn-onlinebanking-giropay
- btn-onlinebanking-ideal
- btn-onlinebanking-eps
- btn-onlinebanking-post
- btn-onlinebanking- eprzelewy
- btn-card-myone
- btn-card-bonuscard
- btn-card-postcard
- btn-card-lasercard
- btn-directdebit-zvt
- btn-directdebit-intercardelv
- btn-directdebit-billpay
- btn-invoice-billpay
- btn-wallet-masterpass

7.5.3 Spécifique pour formulaire de cartes

- page-card
 - page-creditcard-visa
 - page-creditcard-mastercard
 - page-creditcard-maestro
 - page-creditcard-amex
 - page-creditcard-jcb
 - page-creditcard-dinersclub
 - page-creditcard-saferpay
 - page-card-myone
 - page-card-bonuscard
 - page-card-postcard
 - page-card-lasercard
- text-hint
- icon-hint

7.5.4 Spécifique pour recouvrement direct

- page-directdebit
 - page-directdebit-billpay
 - page-directdebit-intercardelv
 - page-directdebit-zvt

-

7.5.5 Spécifique pour Online Banking

- page-onlinebanking
 - page-onlinebanking-ideal
 - page-onlinebanking-giropay

7.5.6 Spécifique pour formulaire de facture

- page-invoice
 - page-invoice-billpay

7.5.7 Spécifique pour formulaire d'adresse (y compris formulaire d'adresse Billpay)

- page-address
- form-col-small
- form-col-large

7.5.8 Spécifique pour CG

- page-termsandconditions

7.5.9 Spécifique pour pages Redirect (y compris 3D-Secure)

- page-redirect

7.5.10 Spécifique pour pages d'erreur

- page-error

7.5.11 Spécifique pour Confirmation Page

- page-confirmation

8 Environnement de test Saferpay

En vue de la phase d'intégration et afin de tester Saferpay, nous mettons à votre disposition notre environnement de test externe (ETU).

Vous pouvez tester Saferpay à l'aide de simulateurs pour tous les moyens de paiement courants sur votre propre compte de test en dehors de votre environnement de production.

Toutes les informations sur notre environnement de test peuvent être trouvées à l'adresse suivante:

<https://www.six-payment-services.com/en/site/saferpay-support/testaccount.html>

9 Exemples

9.1 Remarque importante



Veillez à ce que vos propres valeurs soient encodées en HTML (Entityou Unicode). Ainsi vous êtes sûrs que tous les caractères spéciaux seront correctement transmis à Saferpay.

9.2 C# avec la .NET LIB

Création d'un lien de paiement avec CreatePayInit :

```
MessageFactorymf = new MessageFactory();
mf.Open(""); // Saferpay configuration path, e.g. "c:\\Programme\\Saferpay\\Client"
mo_payinit = mf.CreatePayInit();

string m_accountid = "401860-17795278";
string m_amount = "2095";
string m_currency= "EUR";
string m_description = "Test Einkauf";
string m_address = "no";
string m_orderid= "0815-4711";
string m_backlink= "http://www.myshop.com/back.aspx";
string m_faillink= "http://www.myshop.com/Fail.aspx";
string m_successlink = "http://www.myshop.com/Success.aspx";
string m_notifyurl = "http://www.myshop.com/notify.aspx";

mo_payinit.SetAttribute("ACCOUNTID", m_accountid);
mo_payinit.SetAttribute("AMOUNT", m_amount);
mo_payinit.SetAttribute("CURRENCY", m_currency);
mo_payinit.SetAttribute("DELIVERY", m_address);
mo_payinit.SetAttribute("ORDERID", Server.HtmlEncode(m_orderid));
mo_payinit.SetAttribute("DESCRIPTION", Server.HtmlEncode(m_description));
mo_payinit.SetAttribute("SUCESSLINK", m_successlink);
mo_payinit.SetAttribute("BACKLINK", m_backlink);
mo_payinit.SetAttribute("FAILLINK", m_faillink);
mo_payinit.SetAttribute("NOTIFYURL", m_notifyurl);

string paymenturl = mo_payinit.getURL();
```

Vérification du message PayConfirm :

```
string data = Request.QueryString.Get("DATA");
string signature = Request.QueryString.Get("SIGNATURE");

MessageFactorymf = new MessageFactory();
mf.Open("");
mo_payconfirm = mf.VerifyPayConfirm(data, signature);

string id = mo_payconfirm.GetAttribute(ID);
```

Comptabilisation du paiement avec CreatePayComplete :

```
MessageFactorymf = new MessageFactory();
mf.Open("");
MessageObject mo_paycomplete = mf.CreateRequest("PayComplete");

mo_paycomplete.SetAttribute("ID", id);
mo_paycomplete.SetAttribute("ACCOUNTID", m_accountid);

MessageObjectcaptureresponse = mo_paycomplete.Capture();
```


Réponse PayComplete :

```
intresult = Convert.ToInt32(response.GetAttribute("RESULT"));
if (result == 0)
{
    String id = captureresponse.GetAttribute("ID");
    String msg = captureresponse.GetAttribute("MESSAGE");
    Console.WriteLine("Verbuchungerfolgreich! ");
}
else
{
    Console.WriteLine("Verbuchungfehlgeschlagen! ");
    return;
}
```

9.3 Java avec la Java LIB

Création d'un lien de paiement avec CreatePayInit :

```
import Saferpay.*
import org.apache.commons.lang.*

MessageFactorymf = new MessageFactory();
mf.Open(""); // Saferpay configuration path, e.g. "c:\\Programme\\Saferpay\\Client"
MessageObjectmo_payinit = mf.CreatePayInit();

Stringm_accountid = "401860-17795278";
Stringm_amount = "2095";
Stringm_currency = "EUR";
Stringm_description = "Test Einkauf";
string m_address = "no";
Stringm_orderid = "0815-4711";
Stringm_backlink = "http://www.myshop.com/back.jsp";
Stringm_faillink = "http://www.myshop.com/Fail.jsp";
Stringm_successlink = "http://www.myshop.com/Success.jsp";
Stringm_notifyurl = "http://www.myshop.com/notify.jsp";

mo_payinit.SetAttribute("ACCOUNTID", m_accountid);
mo_payinit.SetAttribute("AMOUNT", m_amount);
mo_payinit.SetAttribute("CURRENCY", m_currency);
mo_payinit.SetAttribute("DELIVERY", m_address);
mo_payinit.SetAttribute("ORDERID", StringEscapeUtils.escapeHtml(m_orderid));
mo_payinit.SetAttribute("DESCRIPTION", StringEscapeUtils.escapeHtml(m_description));
mo_payinit.SetAttribute("SUCCESSLINK", m_successlink);
mo_payinit.SetAttribute("BACKLINK", m_backlink);
mo_payinit.SetAttribute("FAILLINK", m_faillink);
mo_payinit.SetAttribute("NOTIFYURL", m_notifyurl);

StringpaymentURL = mo_payinit.getURL();
```

Vérification du message PayConfirm :

```
import Saferpay.*

String data = request.getParamètres("DATA");
String signature = request.getParamètres("SIGNATURE");

MessageFactorymf = new MessageFactory();
mf.Open("");
MessageObjectmo_payconfirm = mf.VerifyPayConfirm(data, signature);

Stringid = mo_payconfirm.GetAttribute(ID);
```

Comptabilisation du paiement avec CreatePayComplete :

```
MessageFactorymf = new MessageFactory();
mf.Open("");
MessageObjectmo_paycomplete = mf.CreatePayComplete(ID, "");

mo_paycomplete.SetAttribute("ID", id);
mo_paycomplete.SetAttribute("ACCOUNTID", m_accountid);

MessageObjectcaptureresponse = mo_paycomplete.Capture();
```

Réponse PayComplete :

```
intresult = response.GetAttribute("RESULT");
if (result == 0)
{
    String id = captureresponse.GetAttribute("ID");
    String msg = captureresponseresponse.GetAttribute("MESSAGE");
    System.out.println("Verbuchungserfolgreich! ");
}
else
{
    System.out.println("Verbuchungsfehlgeschlagen! ");
    return;
}
```

9.4 Exécution d'une ligne de commande avec la Java LIB

Création d'un lien de paiement avec CreatePayInit :

```
java -jar Saferpay.jar -payinit -p C:\Programme\Saferpay\Client -a AMOUNT 1930 -a CURRENCY EUR -a
DELIVERY no -a ACCOUNTID 401860-17795278 -a DESCRIPTION "Test Einkauf" -a ORDERID 0815-4711 -a
FAILLINK "http://www.testshop.de/fail.php" -a SUCCESSLINK "http://www.testshop.de/success.php"
-a BACKLINK "http://www.testshop.de/back.php" -a NOTIFYURL "http://www.testshop.de/log.php"
```

Lien de paiement créé :

```
https://test.saferpay.com/vt2/Pay.aspx?
DATA=%3CIDP+ALLOWCOLLECT%3D%22no%22+EXPIRATION%3D%2220110325+16%3A06%3A53%22+DESCRIPTION%3D%22
Test+Einkauf%22+BACKLINK%3D%22http%3A%2F%2Fwww.testshop.de%2Fback.php%22+AMOUNT%3D%221930%22+D
ELIVERY%3D%22no%22+ACCOUNTID%3D%22401860-17795278%22+SUCCESSLINK%3D%22http%3A%2F%2Fwww.testsho
p.de%2Fsuccess.php%22+CURRENCY%3D%22EUR%22+ORDERID%3D%220815-4711%22+FAILLINK%3D%22http%3A%2F%
2Fwww.testshop.de%2Ffail.php%22+MSGTYPE%3D%22PayInit%22+KEYID%3D%220-99867-959af13b7be94dd99cb
d20ac7caa5888%22+NOTIFYURL%3D%22http%3A%2F%2Fwww.testshop.de%2Flog.php%22+TOKEN%3D%228ae94d505
62c01e0d24085728e033992%22%2F%3E&SIGNATURE=38FE7F5483578D24EDC3F9A8D00CC1B87733B5CE1F984F82E2E
D084C52B01B0E783C8007F022610D93E26E2AA254C02245A09A0F25A0C5A4961B9FDFB9FF8F50
```

Vérification du message PayConfirm :

Renvoi vers la boutique en ligne et le SUCCESSLINK après paiement réussi :

```
http://www.testshop.de/success.php?
DATA=%3CIDP+MSGTYPE%3d%22PayConfirm%22+TOKEN%3d%22%28unused%29%22+VTVERIFY%3d%22%28obsolete%29%22+KEYID%3d%221-0%22+ID%3d%22brlb01AS3AphUA4fnAr0bQUS7thA%22+ACCOUNTID%3d%22401860-17795278%22+PROVIDERID%3d%2290%22+PROVIDERNAME%3d%22Saferpay+Test+Card%22+ORDERID%3d%220815-4711%22+AMOUNT%3d%221930%22+CURRENCY%3d%22EUR%22+IP%3d%22193.247.180.193%22+IPCOUNTRY%3d%22CH%22+CCOUNTRY%3d%22XX%22+MPI_LIABILITYSHIFT%3d%22yes%22+MPI_XID%3d%22NEpiXQIKWz8xBBszF38FVUUwTAo%3d%22+ECI%3d%222%22+XID%3d%22NEpiXQIKWz8xBBszF38FVUUwTAo%3d%22+%2f%3E&SIGNATURE=b43f0ac94ce260fa520010558dc552ef1c67a839538bd2346d6e593fa74b3e2db22e6de0f494a312bf3436af662219144dda2bf472a5447da205009668a791fa
```

DATA reçue :

```
<IDP MSGTYPE="PayConfirm" TOKEN="(unused)" VTVERIFY="(obsolete)" KEYID="1-0"
ID="brlb01AS3AphUA4fnAr0bQUS7thA" ACCOUNTID="401860-17795278" PROVIDERID="90"
PROVIDERNAME="Saferpay Test Card" ORDERID="0815-4711" AMOUNT="1930" CURRENCY="EUR"
IP="193.247.180.193" IPCOUNTRY="CH" CCOUNTRY="XX" MPI_LIABILITYSHIFT="yes"
MPI_XID="NEpiXQIKWz8xBBszF38FVUUwTAo=" ECI="2" XID="NEpiXQIKWz8xBBszF38FVUUwTAo=" />
```

SIGNATURE reçue :

```
b43f0ac94ce260fa520010558dc552ef1c67a839538bd2346d6e593fa74b3e2db22e6de0f494a312bf3436af662219144dda2bf472a5447da205009668a791fa
```

Réalisation du VerifyPayConfirm:

```
java -jar Saferpay.jar -payconfirm -p C:\Programme\Saferpay\Client -d
%3CIDP+MSGTYPE%3d%22PayConfirm%22+TOKEN%3d%22%28unused%29%22+VTVERIFY%3d%22%28obsolete%29%22+KEYID%3d%221-0%22+ID%3d%22brlb01AS3AphUA4fnAr0bQUS7thA%22+ACCOUNTID%3d%22401860-17795278%22+PROVIDERID%3d%2290%22+PROVIDERNAME%3d%22Saferpay+Test+Card%22+ORDERID%3d%220815-4711%22+AMOUNT%3d%221930%22+CURRENCY%3d%22EUR%22+IP%3d%22193.247.180.193%22+IPCOUNTRY%3d%22CH%22+CCOUNTRY%3d%22XX%22+MPI_LIABILITYSHIFT%3d%22yes%22+MPI_XID%3d%22NEpiXQIKWz8xBBszF38FVUUwTAo%3d%22+ECI%3d%222%22+XID%3d%22NEpiXQIKWz8xBBszF38FVUUwTAo%3d%22+%2f%3E
-sb43f0ac94ce260fa520010558dc552ef1c67a839538bd2346d6e593fa74b3e2db22e6de0f494a312bf3436af662219144dda2bf472a5447da205009668a791fa
```

Comptabilisation du paiement avec CreatePayComplete :

```
java -jar Saferpay.jar -capture -p C:\Programme\Saferpay\Client -i brlb01AS3AphUA4fnAr0bQUS7thA
-a ACCOUNTID 401860-17795278-of capt.txt
```

9.5 https Interface

Création d'un lien de paiement avec CreatePayInit :

```
https://test.saferpay.com/hosting/CreatePayInit.asp?  
ACCOUNTID=99867-94913159&ORDERID=123456789-001&AMOUNT=1000&CURRENCY=EUR&DESCRIPTION=Testkauf&S  
UCCESSLINK="http://www.myshop.com/Success.aspx"&FAILLINK="http://www.myshop.com/Fail.aspx"&BAC  
KLINK="http://www.myshop.com/back.aspx"&NOTIFYURL="http://www.myshop.com/notify.aspx"
```

Réponse fournit l'URL de paiement :

```
https://test.saferpay.com/vt2/Pay.aspx?  
DATA=%3cIDP+MSGTYPE%3d%22PayInit%22+MSG_GUID%3d%22939a4c930b5c482588d91f54f74ac110%22+CLIENTVE  
RSION%3d%222.0%22+KEYID%3d%220-99867-7d5a273c0f5043e28811e764d6433086%22+TOKEN%3d%22bbf6577cd8  
e74d65a27f084c9cfe2592%22+ALLOWCOLLECT%3d%22no%22+DELIVERY%3d%22no%22+EXPIRATION%3d%220110625  
+12%3a01%3a56%22+ACCOUNTID%3d%2299867-94913159%22+AMOUNT%3d%221000%22+CURRENCY%3d%22EUR%22+DES  
CRIPTION%3d%22Testkauf%22+SUCCESSLINK%3d%22http%3a%2f%2fwww.myshop.com%2fSuccess.aspx%22+BACKL  
INK%3d%22http%3a%2f%2fwww.myshop.com%2fback.aspx%22+FAILLINK%3d%22http%3a%2f%2fwww.myshop.com%  
2fFail.aspx%22+ORDERID%3d%22123456789-001%22+CCNAME%3d%22yes%22+NOTIFYURL%3d%22http%3a%2f%2fw  
w.myshop.com%2fnotify.aspx%22+%2f%3e&SIGNATURE=428b356c87f7fdcf44417f670197c4a6395385e623e2246  
53610b94db8acclead509b7b5e6dfc465dcf987b3cf4b284fc799ee93ff9cb151c3bc9981e8320232
```

Vérification du message PayConfirm:

Renvoi vers la boutique en ligne et le SUCCESSLINK après autorisation réussie :

```
http://www.myshop.com/Success.aspx?  
DATA=%3CIDP+MSGTYPE%3d%22PayConfirm%22+TOKEN%3d%22%28unused%29%22+VTVERIFY%3d%22%28obsolete%29  
%22+KEYID%3d%221-0%22+ID%3d%22A668MSAprOj4tAzv7G9lAQufUr3A%22+ACCOUNTID%3d%2299867-94913159%22  
+PROVIDERID%3d%2290%22+PROVIDERNAME%3d%22Saferpay+Test+Card%22+ORDERID%3d%22123456789-001%22+A  
MOUNT%3d%221000%22+CURRENCY%3d%22EUR%22+IP%3d%22193.247.180.193%22+IPCOUNTRY%3d%22CH%22+CCOUN  
TRY%3d%22XX%22+MPI_LIABILITYSHIFT%3d%22yes%22+MPI_TX_CAVV%3d%22AAABBIIFmAAAAAAAAAAAAAAAAAAAA%3d%  
22+MPI_XID%3d%22CxMTYwhoUXtCBAEndBULcRIQaAY%3d%22+ECI%3d%221%22+CAVV%3d%22AAABBIIFmAAAAAAAAAAAA  
AAAAAAAA%3d%22+XID%3d%22CxMTYwhoUXtCBAEndBULcRIQaAY%3d%22+%2f%3e&SIGNATURE=7b2bb163f4ef86d969d9  
92b4e2d61ad48d3b9022e0ec68177e35fe53184e6b3399730d1a3641d2a984ce38699daad72ab006d5d6a9565c5ae1  
cff8bdc8a1eb63
```

DATA reçue :

```
<IDP MSGTYPE="PayConfirm" TOKEN="(unused)" VTVERIFY="(obsolete)" KEYID="1-0"  
ID="A668MSAprOj4tAzv7G9lAQufUr3A" ACCOUNTID="401860-17795278" PROVIDERID="90"  
PROVIDERNAME="Saferpay Test Card" ORDERID="123456789-001" AMOUNT="1000" CURRENCY="EUR"  
IP="193.247.180.193" IPCOUNTRY="CH" CCCOUNTRY="XX" MPI_LIABILITYSHIFT="yes"  
MPI_TX_CAVV="AAABBIIFmAAAAAAAAAAAAAAAAAAAA=" MPI_XID="CxMTYwhoUXtCBAEndBULcRIQaAY=" ECI="1"  
CAVV="AAABBIIFmAAAAAAAAAAAAAAAAAAAA=" XID="CxMTYwhoUXtCBAEndBULcRIQaAY=" />
```

SIGNATURE reçue :

```
7b2bb163f4ef86d969d992b4e2d61ad48d3b9022e0ec68177e35fe53184e6b3399730d1a3641d2a984ce38699daad7  
2ab006d5d6a9565c5ae1cff8bdc8a1eb63
```

Exécution httpsVerifyPayConfirm:

```
https://test.saferpay.com/hosting/VerifyPayConfirm.asp?
spPassword=XAjc3Kna&ACCOUNTID=99867-94913159&DATA=%3CIDP+MSGTYPE%3d%22PayConfirm%22+TOKEN%3d%2
2%28unused%29%22+VTVERIFY%3d%22%28obsolete%29%22+KEYID%3d%221-0%22+ID%3d%22A668MSAprOj4tAzv7G9
1AQUfUr3A%22+ACCOUNTID%3d%2299867-94913159%22+PROVIDERID%3d%2290%22+PROVIDERNAME%3d%22Saferpay
+Test+Card%22+ORDERID%3d%22123456789-001%22+AMOUNT%3d%221000%22+CURRENCY%3d%22EUR%22+IP%3d%221
93.247.180.193%22+IPCOUNTRY%3d%22CH%22+CCCOUNTRY%3d%22XX%22+MPI_LIABILITYSHIFT%3d%22yes%22+MPI
_TX_CAVV%3d%22AAABBIIFmAAAAAAAAAAAAAAAAAAAA%3d%22+MPI_XID%3d%22CxMTYwhoUXtCBAEndBULcRIQaAY%3d%22
+ECI%3d%221%22+CAVV%3d%22AAABBIIFmAAAAAAAAAAAAAAAAAAAA%3d%22+XID%3d%22CxMTYwhoUXtCBAEndBULcRIQaA
Y%3d%22+%2f%3E&SIGNATURE=7b2bb163f4ef86d969d992b4e2d61ad48d3b9022e0ec68177e35fe53184e6b3399730
d1a3641d2a984ce38699daad72ab006d5d6a9565c5ae1cff8bdc8a1eb63
```

Réponse fournit l'identifiant de la transaction :

```
OK:ID=A668MSAprOj4tAzv7G91AQUfUr3A&TOKEN=(unused)
```

Comptabilisation du paiement avec CreatePayComplete :

```
https://test.saferpay.com/hosting/PayCompleteV2.asp?
spPassword=XAjc3Kna&ACCOUNTID=401860-17795278&ID=A668MSAprOj4tAzv7G91AQUfUr3A
```

Réponse fournit le résultat :

```
OK:<IDP RESULT="0"/>
```

10 Codes d'erreur

Les codes de réponse suivants peuvent être transmis en retour via la Saferpay Library :

Code	Nom	Description
Erreurs générales		
0x80040201	ContextMissing	Pas de contexte de configuration défini. Exécutez Open() avant d'activer d'autres méthodes.
0x80040202	File Not Found	Fichier n'a pas pu être trouvé. Vérifiez si les composants sont installés correctement.
0x80040203	File Access Error	Erreur lors de l'accès au fichier. Vérifiez les droits d'accès dans le dossier de configuration.
0x80040204	Invalid File Format	Fichier ne contient pas de contenu XML valide.
0x80040206	InvalidPath	Chemin ou URL non valide.
0x80040207	Invalid Option	Saisie de paramètres non valide.
0x80040208	RequestFailed	Echec de la demande.
0x80040209	CryptographicError	Erreur de codage.
0x8004020f	No Configuration	Aucune donnée de configuration existante dans le chemin d'accès de Open().
0x80040214	VerifyFailed	Signature non valide. Echec de la vérification de la signature.
Erreurs HTTP		
0x80042019	Invalid URL	Saisie invalide de l'URL.
0x80042021	DNS Error	Erreur lors de la résolution d'une adresse.
0x80042xxx	HTTP Server Error	Erreur sur le serveur HTTP. Les trois derniers chiffres correspondent au code de statut renvoyé par le serveur.
Erreur de socket		
0x80043xxx	Socket Base	Les trois derniers chiffres correspondent au code d'erreur de socket.

11 Contact

11.1 Saferpay Integration Team

Avez-vous des questions concernant ce document, rencontrez-vous des problèmes lors de l'intégration de Saferpay ou avez-vous besoin d'aide? Dans ce cas n'hésitez pas à contacter notre Integration Team:

Saferpay Suisse
SIX Payment Services SA
Hardturmstrasse 201
CH-8021 Zurich
+41 848 66 44 44
www.six-payment-services.com/saferpay
integration.saferpay@six-payment-services.com

Saferpay Europe
SIX Payment Services (Germany) GmbH
Langenhorner Chaussee 92-94
D-22415 Hambourg
+49 40 325 967 280
www.six-payment-services.com/saferpay
integration.saferpay@six-payment-services.com

11.2 Saferpay Support Team

Avez-vous des questions relatives aux messages d'erreur ou rencontrez-vous des problèmes en cours d'utilisation? Dans ce cas notre Support Team est à votre disposition:

Saferpay Suisse
SIX Payment Services SA
Hardturmstrasse 201
CH-8021 Zurich
+41 848 66 44 44
www.six-payment-services.com/saferpay
support.saferpay@six-payment-services.com

Saferpay Europe
SIX Payment Services (Germany) GmbH
Langenhorner Chaussee 92-94
D-22415 Hambourg
+49 40 325 967 250
www.six-payment-services.com/saferpay
support.saferpay@six-payment-services.com

Toute l'équipe de Saferpay vous souhaite beaucoup de succès avec la solution de paiement en ligne Saferpay!