



Saferpay E-Commerce Hosting Schnittstelle

Datum: **August 2006**
Version: **1.4**
Status: **Released**

INHALTSVERZEICHNIS

1	<u>EINLEITUNG</u>	3
1.1	VORAUSSETZUNGEN	3
1.2	UNTERSTÜTZTE FUNKTIONALITÄTEN	3
1.3	EINSCHRÄNKUNGEN	3
1.4	SICHERHEIT	4
2	<u>SERVER-TO-SERVER SCHNITTSTELLE</u>	5
2.1	VORTEILE	5
2.2	GATEWAY ADRESSEN	6
2.3	ABLAUF	6
2.4	ERZEUGUNG DER PAYINIT URL (CREATEPAYINIT)	6
2.5	PRÜFUNG VON PAYCONFIRM (VERIFYPAYCONFIRM)	7
2.5.1	BEISPIEL VERIFYPAYCONFIRM	7
2.6	FREIGABE ZUR ZAHLUNG (PAYCOMPLETE)	7
3	<u>REDIRECT SCHNITTSTELLE</u>	8
3.1	NACHTEILE	8
3.2	GATEWAY ADRESSEN	8
3.3	ABLAUF	8
3.4	REDIRECT VON PAYINIT	9
3.5	REDIRECT VON PAYCONFIRM	9
4	<u>PARAMETER</u>	10
4.1	ZAHLUNGSLINK	10
4.2	SUCCESSLINK	10
5	<u>INTEGRATION UND PROGRAMMIERUNG</u>	12
5.1	VORAUSSETZUNGEN	12
5.1.1	VERARBEITUNG VON KLARTEXT	12
5.1.2	STEUERUNG DES VIRTUELLEN SAFERPAY TERMINALFENSTERS	13
5.2	KLARTEXT MELDUNGEN	14
5.2.1	AUFRUF VON CREATEPAYINIT.ASP	14
5.2.2	VERWENDUNG VON VERIFYPAYCONFIRM.ASP	14
5.2.3	AUFRUF VON PAYCOMPLETE.ASP	15
5.3	EINBETTUNG IN HTML	16
5.3.1	FORM (POST)	16
5.3.2	LINK (GET)	16
5.3.3	BUTTON	16

1 EINLEITUNG

In diesem Dokument werden die Schnittstellen des Saferpay-Zertifikat Hosting-Servers beschrieben. Wenn in der Folge von dem Saferpay Gateway die Rede ist, ist damit der Zertifikats-Server gemeint.

Sinnvollerweise kann der Saferpay Gateway dann eingesetzt werden, wenn auf dem Zielsystem die Saferpay Client Komponente nicht installiert oder betrieben werden kann. Dies mag bei einigen ISP (Internet Service Providern) der Fall sein, die Installationen von Fremdsoftware nicht erlauben.

1.1 VORAUSSETZUNGEN

Um den Saferpay Gateway nutzen zu können, müssen die Händlerdaten mit SUCCESSLINK, FAILLINK und BACKLINK auf dem Gateway eingerichtet werden.

Die für den Händler erzeugten Saferpay Zertifikate werden auf dem Saferpay Gateway gehostet.

1.2 UNTERSTÜTZTE FUNKTIONALITÄTEN

Der Saferpay Gateway kann folgende Aufgaben übernehmen:

- Erzeugung von gesicherten, digital signierten Saferpay URLs (CreatePayInit)
- Überprüfung von digital signierten Saferpay Antworten (VerifyPayConfirm)
- Freigabe von Autorisierungen zur Buchung (PayComplete)
- Die Übergabe von Attributen an den Saferpay Gateway erfolgt per GET oder POST

1.3 EINSCHRÄNKUNGEN

Die folgenden Funktionen werden aus Sicherheitsgründen nur von der Saferpay Client Komponente unterstützt:

- Stornierung von Autorisierungen oder Buchungen,
- Steuerung des Tagesabschlusses,
- Nutzung der „Inquiry“ und „Card Authorization“ Schnittstellen

1.4 SICHERHEIT

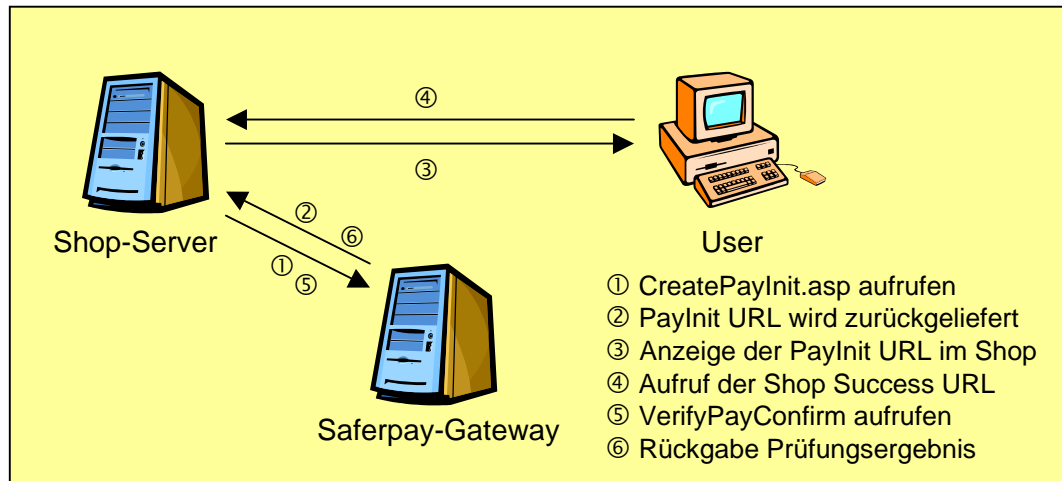
Saferpay steht für Sicherheit im elektronischen Zahlungsverkehr. Dabei stellt Saferpay sicher, dass die mit dem e-Commerce System ausgetauschten Daten nicht manipuliert werden können. Manipulationen, die versierte Internetnutzer vornehmen könnten, werden erkannt und dem e-Commerce System angezeigt.

Der Saferpay Gateway erlaubt es, eine neue Saferpay URL zu generieren, ohne dass die Saferpay Komponente oder die Saferpay-Zertifikate auf dem Shop-Server installiert sind. Damit wird aber auch der Weg für andere Internet-Nutzer geöffnet, neue Saferpay URLs im Namen des Händlers zu erzeugen.

Dies stellt insofern keine Sicherheitslücke dar, da nur Verkäufe im Namen des Händlers getätigt werden können. Es ist nicht möglich, über diesen Weg Gutschriften oder Stornierungen auf dem Händlerkonto vorzunehmen.

2 SERVER-TO-SERVER SCHNITTSTELLE

In dieser Variante findet eine Server-to-Server Kommunikation statt: Der Shop-Server sendet Anfragen per http/GET an den Hosting Gateway, der eine Antwort als „Klartext“ zurückliefert.



Zum Erzeugen eines Zahlungslinks sendet der Shop-Server die jeweiligen Zahlungsattribute wie Betrag, Währung, Verkaufsbeschreibung und Saferpay-Account, an den Hosting Gateway (1). Der Gateway erzeugt den Zahlungslink und liefert ihn als Textausgabe zurück (2).

Dem Kunden wird der Zahlungslink z.B. als [Bezahlung] Button angezeigt (3). Nach der Zahlung wird der Kunde wieder zum Shop zurückgelenkt (4).

Nach einer erfolgreichen Zahlung können die zurückerhaltenen Zahlungsdaten verifiziert werden. Der Shop-Server sendet die Daten an den Hosting Gateway zur Überprüfung (5). Der Gateway liefert prompt das Ergebnis der Überprüfung zurück (6).

Alternativ kann die Schnittstelle auch über HTTPS angesteuert werden. Diese Variante macht Sinn, wenn die Anfrage direkt vom Webserver kommt. Für Webshops, bei denen der Warenkorb durch Client-seitiges JavaScript erzeugt wird, sollte aus Gründen der Performance mit der http-Schnittstelle gearbeitet werden (ca. eine Sekunde schneller).

Die Übergabe der Attribute an den Saferpay Gateway erfolgt mittels GET oder POST.

2.1 VORTEILE

Das Erzeugen des Zahlungslinks und das spätere Auswerten der Antwortdaten erfolgt im Shop-Server, ohne dass dies der Kunde mitbekommt.

Die PayConfirm Antwortdaten werden direkt an den Webshop zurückgesendet und erst dort über den Saferpay Gateway geprüft.

Die Freigabe einer Zahlung erfolgt durch den Aufruf von PayComplete.

2.2 GATEWAY ADRESSEN

Der Saferpay Gateway wird über folgende Webadressen angesteuert:

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

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

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

2.3 ABLAUF

Grundsätzlich läuft eine Transaktion nach folgendem Schema ab:

- 1) Erzeugen der Saferpay-URL (CreatePayInit)
- 2) Die Autorisierung wird von Kunden unter Saferpay abgewickelt.
- 3) Als Antwort wird der SUCCESSLINK mit dem Resultat aufgerufen
- 4) Die Antwort sollte auf Plausibilität/Manipulation geprüft werden (VerifyPayConfirm).
- 5) Die Autorisierung muss verbucht werden entweder manuell über das Saferpay Backoffice oder automatisiert (PayComplete).

2.4 ERZEUGUNG DER PAYINIT URL (CREATEPAYINIT)

Das e-Commerce System übermittelt die Saferpay Attribute an den Saferpay Gateway (per GET oder POST). Der Gateway erzeugt eine digital signierte PayInit URL und liefert das Ergebnis im Klartext sofort wieder zurück. Die so erzeugte PayInit URL wird auf der Website eingefügt, z.B. als „Bezahlen“ Link.

Beispiel

Die Attribute werden hier per GET an den Saferpay Gateway gesendet:

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

Das Ergebnis wird im Klartext ohne HTML Tags zurückgeliefert:

```
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%20DELIVERY%3d%22no%22%20DESCRIPTION%3d%22Testkauf%20Warenkorb%22%20EXPIRATION%3d%2220010408%2012%3a13%3a50%22%20FAILLINK%3d%22%2e%22%20KEYID%3d%220%2d37217%2dea645c3f3f0911d583d70050da413f31%22%20MSGTYPE%3d%22PayInit%22%20SUCCESSLINK%3d%22%2e%22%20TOKEN%3d%22ea645c5d3f0911d583d70050da413f31%22%2f%3e&SIGNATURE=2f1ec1fa51002817941c22e98b9047422ba9ff8fce8b61dab8208a5aa8c82be7cda02ff8a66930481fc19b16d05e7bcedd2b0e5be98fecad3d48bd43916a502f
```

Im Falle eines Fehlers wird die Kennung „ERROR“ zusammen mit einer Fehlerbeschreibung zurückgeliefert:

ERROR: Missing AMOUNT attribute

2.5 PRÜFUNG VON PAYCONFIRM (VERIFYPAYCONFIRM)

Nach dem gleichen Schema wie zur Erzeugung der PayInit URL kann auch mit dem Schritt PayConfirm verfahren werden. Bei Aufruf des SUCCESSLINKs werden die Parameter DATA und SIGNATURE an das E-Commerce System zurückgeliefert.

Zur Überprüfung von DATA werden die beiden Parameter an den Saferpay Gateway gesendet, der mit OK oder ERROR antwortet.

2.5.1 BEISPIEL VERIFYPAYCONFIRM

Die Parameter DATA und SIGNATURE werden per GET oder POST an den Saferpay Gateway gesendet:

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

Stimmt die digitale Signatur mit den Werten aus DATA überein, wird die positive Überprüfung mit OK unter Angabe der Saferpay ID und TOKEN angezeigt:

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

Im Falle eines Fehlers wird der Text "ERROR" zusammen mit einer Fehlerbeschreibung zurückgeliefert:

ERROR: Possible manipulation

In DATA sind alle Felder der Antwortnachricht XML kodiert enthalten, wie z.B. PROVIDERID, PROVIDERNAME, usw.

2.6 FREIGABE ZUR ZAHLUNG (PAYCOMPLETE)

Das Verbuchen einer Autorisierung erfolgt über den Aufruf der PayComplete URL des Gateways. Es müssen die Attribute ACCOUNTID und ID per GET oder POST an den Saferpay Gateway gesendet werden.

Alle anderen Attribute, wie ACTION oder AMOUNT werden ignoriert.

Der Ausführungsstatus von PayComplete wird vom Saferpay Gateway mit OK oder ERROR als Klartext beantwortet.

Beispiel

Aufruf von PayComplete zur Freigabe einer Buchung (GET):

<https://www.saferpay.com/hosting/PayComplete.asp?ACCOUNTID=99867-94913159&ID=5sfhmkq3rg54345abcd234>

Konnte die Operation erfolgreich durchgeführt werden, wird „OK“ zurückgeliefert:

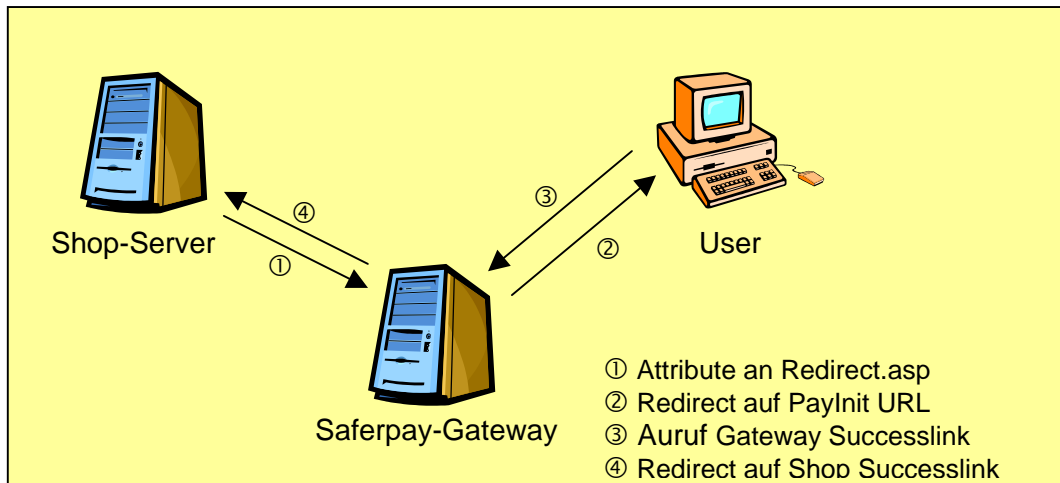
OK

Im Falle eines Fehlers wird der Text „ERROR“ zusammen mit einer Fehlerbeschreibung zurückgeliefert:

ERROR: Error description

3 REDIRECT SCHNITTSTELLE

Ist der Webserver nicht in der Lage, die Saferpay URL per HTTP Request zu erzeugen bzw. die Antwortdaten per HTTP Request zu überprüfen, kann der Saferpay Gateway per „redirect“ genutzt werden.



3.1 NACHTEILE

Die Zahlungsdaten und Attribute werden auf der Website hinterlegt und können so vom Internet-Nutzer manipuliert werden. Ebenso lässt sich die PayConfirm-Nachricht verändern oder simulieren.

3.2 GATEWAY ADRESSEN

Der Saferpay Gateway wird über die Webadresse

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

angesteuert.

3.3 ABLAUF

Grundsätzlich läuft eine Transaktion per Redirect nach folgendem Schema ab:

Step 1: Aufruf von Redirect, um indirekt Saferpay aufzurufen

Step 2: Die Autorisierung wird von Kunden unter Saferpay abgewickelt

Step 3: Als Antwort wird der SUCCESSLINK mit dem Resultat aufgerufen

Step 4: Die Autorisierung muss entweder manuell über das Saferpay Backoffice oder automatisiert (PayComplete) verbucht werden.

3.4 REDIRECT VON PAYINIT

Die zur Zahlung notwendigen Attribute werden per GET oder POST an den Saferpay Gateway übermittelt. Der Gateway erzeugt eine Saferpay-URL, die unmittelbar per „redirect“ aufgerufen wird.

Im Attribut SUCCESSLINK können zusätzliche Parameter eingefügt werden wie z.B. eine Session-ID.

3.5 REDIRECT VON PAYCONFIRM

Die Autorisierungsantwort des Saferpay Servers (PayConfirm) wird zunächst an den Saferpay Gateway übermittelt. Der Gateway prüft die Antwort auf Plausibilität und Manipulation (VerifyPayConfirm) und übermittelt das Ergebnis an den ursprünglichen SUCCESSLINK. Dies geschieht automatisch durch den Aufruf von „Redirect.asp“.

Das Ergebnis der Prüfung wird im Attribut RESULT angezeigt, das entweder „1“ für Erfolg oder „0“ bei Manipulation oder Fehler enthält.

An den SUCCESSLINK die von Saferpay zurückgelieferten Attribute angehängt. Dabei handelt es sich um ID, TOKEN, AMOUNT, CURRENCY, PROVIDERID, PROVIDERNAME, ACCOUNTID und optional um ORDERID. Im Fehlerfall werden diese Saferpay Attribute nicht zurückgeliefert.

Die Übergabe der Ergebnisattribute erfolgt per GET.

Beispiel

Die Attribute werden hier per GET an den Saferpay Gateway gesendet:

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

Der Browser des Internet-Users wird per „Redirect“ sofort auf die Saferpay-Site weitergeführt. Dort findet die Autorisierung über ein beliebiges Zahlungsmittel statt.

Bei positivem Ausgang der Autorisierung wird die Success-Seite auf dem Saferpay Gateway aufgerufen. Die Prüfung der digitalen Signatur findet dort statt.

Die Ergebnisattribute werden per GET an den SUCCESSLINK weitergeleitet, wobei das positive Ergebnis der Überprüfung mit „RESULT=0“ angezeigt wird:

http://www.shop.de/kasse_ok.pl?Session=123&RESULT=0&ID=fasmm...

Im Falle eines Fehlers wird das Feld „RESULT“ mit einem Wert ungleich 0 zurückgeliefert:

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

4 PARAMETER

Alle Parameter sind vollständig im "Saferpay Implementation Guide" aufgeführt (siehe www.saferpay.com/help).

4.1 ZAHLUNGSLINK

Die folgenden Parameter können bei der Erzeugung des Zahlungslinks verwendet werden.

Attribut	Beschreibung
ACCOUNTID	Die Saferpay Kontonummer des Händlers Beispiel: 99867-94913159
AMOUNT	Zahlungsbetrag in kleinster Währungseinheit Beispiel: 1295 steht für 12,95 EUR
CURRENCY	3-stelliger Währungscode Beispiel: EUR, USD, GBP, CHF, etc.
DESCRIPTION	Verkaufsbeschreibung (URL encoded) Beispiel: Dies%20ist%20ein%20Einkauf
CCCVC	Optional: Kartenprüfnummer (CVV/CVC2) abfragen, wenn auf „yes“ gesetzt.
CCNAME	Optional: Name des Karten- oder Kontoinhabers abfragen, wenn auf „yes“ gesetzt
ORDERID	Optional: Händler Referenznummer, max. 12 Stellen Beispiel: 2307020034
RESPONSEFORMAT	Optional: Formatierung der Antwortdaten Beispiel: XML (default) oder URL
SUCCESSLINK	Optional: Webseite (Shop-URL), die nach einer erfolgreichen Autorisierung aufgerufen werden soll. Nur nötig, wenn die Links nicht fest auf dem Hosting-Server hinterlegt wurden.
BACKLINK	Optional: Webseite (Shop-URL), die beim Abbruch des Kunden aufgerufen wird. Nur nötig, wenn die Links nicht fest auf dem Hosting-Server hinterlegt wurden.
FAILLINK	Optional: Webseite (Shop-URL), die aufgerufen wird, wenn die Zahlung nicht durchgeführt werden konnte. Nur nötig, wenn die Links nicht fest auf dem Hosting-Server hinterlegt wurden.
AUTOCLOSE	Optional: Anzahl Sekunden (0 bis n), nach der der Zahler automatisch an den SUCCESSLINK weitergeleitet wird.

4.2 SUCCESSLINK

Die Antwortdaten sind im DATA Parameter als XML-Datensatz enthalten. Folgende Parameter werden u.a. zurückgeliefert:

Attribut	Beschreibung
ID	Saferpay Transaktionskennung, max. 80 Zeichen. Beispiel: QIU3CUbQQ8IISAAIQY7IAMW2E52A

Attribut	Beschreibung
TOKEN	Optional: Reserviert
PROVIDERID	Interne Kennnummer des Verarbeiters
PROVIDERNAME	Name des Verarbeiters
AUTHCODE	Autorisierungscode des Verarbeiters
ORDERID	Optional: Enthält die gleiche Händlerreferenz wie in der Anfrage-Nachricht
CONTRACTNUMBER	Optional: Vertragsnummer des Providers

5 INTEGRATION UND PROGRAMMIERUNG

Die folgenden Beispiele sind für die ASP Programmierung ausgelegt und sollen die Integration des Safepay-Zertifikat Servers näher erläutern. Sinngemäss gilt die Vorgehensweise auch für andere (objektorientierte) Programmiersprachen.

5.1 VORAUSSETZUNGEN

5.1.1 VERARBEITUNG VON KLARTEXT

Eine wichtige Funktion bildet „GetURL()“, die beim Aufruf der Gateway-Seiten CreatePayInit.asp, VerifyPayConfirm.asp, xmlCreatePayInit.asp und xmlVerifyPayConfirm.asp verwendet wird.

5.1.1.1 ASP BEISPIEL

Die Funktion liefert den (Klartext) Output einer Webseite (strURL) zurück:

```
<%  
Function GetURL (strURL)  
  ' Set oHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")  
  ' oHTTP.SetTimeouts 2000, 2000, 2000, 2000  
  Set oHTTP = Server.CreateObject("Microsoft.XMLHTTP")  
  oHTTP.open "GET", strURL, false  
  oHTTP.send  
  GetURL = oHTTP.ResponseText  
  Set oHTTP = Nothing  
End Function  
%>
```

5.1.1.2 PHP BEISPIEL

Die folgende Zeile liefert den (Klartext) Output einer Webseite (strURL) zurück:

```
$Text = join("", file(strURL));
```

5.1.1.3 VB.NET BEISPIEL

Diese Funktion liefert den (Klartext) Output einer Webseite (strURL) zurück:

```
Imports System.Net  
Imports System.IO  
  
Function GetURL(ByVal url As String) As String  
  Dim request As HttpWebRequest  
  Dim response As HttpWebResponse  
  Dim webstream As StreamReader  
  request = WebRequest.Create(url)  
  response = request.GetResponse()  
  webstream = New StreamReader(response.GetResponseStream())  
  GetUrl = webstream.ReadToEnd()  
  webstream.Close()  
End Function
```

5.1.2 STEUERUNG DES VIRTUELLEN SAFERPAY TERMINALFENSTERS

Binden Sie auf Ihren Webseiten das Script „OpenSaferpayScript.js“ ein. Dieses Script stellt Ihnen die Java Funktion „OpenSaferpayTerminal()“ zur Verfügung, mit dem Sie das automatische Öffnen und Schliessen des Saferpay Bezahlfensters steuern können:

```
<SCRIPT SRC="http://www.saferpay.com/OpenSaferpayScript.js"></SCRIPT>
```

5.2 KLARTEXT MELDUNGEN

5.2.1 AUFRUF VON CREATEPAYINIT.ASP

Das Erzeugen des Saferpay-URL ist denkbar einfach. In diesem Beispiel werden die notwendigen Übergabewerte in der Variablen „Attribute“ zwischengespeichert und an den Saferpay Gateway übergeben:

```
<%
  Gateway = "https://www.saferpay.com/hosting/CreatePayInit.asp"

  Attribute = "AMOUNT=100&ACCOUNTID=99867-94913159&CURRENCY=DEM"
  Attribute = Attribute & "&DESCRIPTION=Testeinkauf Saferpay"
%>
```

Es wird die Funktion „GetURL()“ ausgeführt, um über den Saferpay Gateway die digitale Signature und die eigentliche Saferpay-URL zu erzeugen. Das Ergebnis wird in der Variablen „SaferpayURL“ zwischengespeichert:

```
<% SaferpayURL = GetURL(Gateway & "?" & Attribute) %>
```

Der Inhalt von „SaferpayURL“ wird je nach Verwendungszweck an die Webseite ausgegeben. In diesem Beispiel wird das virtuelle Saferpay Terminal in einem neuen Fenster geöffnet, sobald auf den Button „Zur Kasse...“ geklickt wird:

```
<input type="BUTTON" value="zur Kasse..."
  onClick="OpenSaferpayTerminal('<%=SaferpayURL%>',this,'BUTTON') " >
```

Die Abwicklung der Autorisierung unter Saferpay kann nun erfolgen...

In diesem Beispiel wurden die Attribute SUCCESSLINK, BACKLINK und FAILLINK nicht angegeben. Daher werden die Default-Einträge für diese Attribute aus der jeweiligen Saferpay-Gateway-Konfiguration verwendet.

5.2.2 VERWENDUNG VON VERIFYPAYCONFIRM.ASP

Nach erfolgreicher Abwicklung der Zahlung wird die Webseite aufgerufen, die mit SUCCESSLINK angegeben wurde.

In ähnlicher Weise wie „CreatePayInit.asp“ wird „VerifyPayConfirm.asp“ genutzt, um die im SUCCESSLINK zurückgelieferte digitale Signatur zu prüfen. Zunächst ermitteln wir die von Saferpay per GET gelieferten Felder DATA und SIGNATURE:

```
<%
  DATA      = Request.QueryString("DATA")
  SIGNATURE  = Request.QueryString("SIGNATURE")
%>
```

Die Prüfung der digitalen Unterschrift erfolgt über den Aufruf der Funktion „GetURL()“. Das Ergebnis der Prüfung wird in „Result“ zwischengespeichert:

```
<%
  Gateway = "https://www.saferpay.com/hosting/VerifyPayConfirm.asp"

  Result = GetURL(Gateway & "?DATA=" & DATA & "&SIGNATURE=" & SIGNATURE)
%>
```

Result enthält entweder „OK:....“ oder „ERROR:....“. Wir ermitteln zunächst, ob die Prüfung der digitalen Signatur korrekt ausgeführt werden konnte:

```
<%  
    arrayResult = split(Result, ":", 2, 1)  
    Status = arrayResult(0)  
  
    if Status = "OK" then  
        ' positive Prüfung, weitere Attribute aus Status extrahieren  
        arrayStatus = split(arrayResult(1), "&", -1, 1)  
        ...  
    else  
        ' negatives Ergebnis der Prüfung  
        ....  
    end if  
%>
```

Auf die weitere Verarbeitung soll hier nicht näher eingegangen werden.

5.2.3 AUFRUF VON PAYCOMPLETE.ASP

Um eine Autorisierung als Buchung auf dem Saferpay-Server einzutragen, kann PayComplete.asp aufgerufen werden. Die notwendigen Attribute werden :

```
<%  
    Gateway = "https://www.saferpay.com/hosting/PayComplete.asp"  
  
    Attribute = "ACCOUNTID=99867-94913159"  
    Attribute = Attribute & "&ID=658mac33291245553ajkq3so"  
    Attribute = Attribute & "&TOKEN=2AF5690C23BEF264865A648B5368C882121AA"  
%>
```

Es wird die Funktion „GetURL()“ ausgeführt, um über den Saferpay Gateway die Funktion „PayComplete“ auszuführen. Das (Klartext) Ergebnis der Ausführung wird in der Variablen „Result“ zwischengespeichert:

```
<% Result = GetURL(Gateway & "?" & Attribute) %>
```

Der Inhalt von „Result“ muss bei positiver Ausführung „OK“ enthalten:

```
<%  
    if Result = "OK" then  
        ' PayComplete ausgeführt  
        ...  
    else  
        ' Fehler bei Ausführung  
        ...  
    end if  
%>
```

5.3 EINBETTUNG IN HTML

Das folgenden Scripte verdeutlichen den Aufruf des Saferpay Gateways zur Erzeugung der Saferpay URL.

In diesen Beispielen werden SUCCESSLINK, BACKLINK und FAILLINK nicht verwendet. Stattdessen werde die Inhalte dieser Links vom Saferpay Gateway aus den dort hinterlegten Konfigurationsdaten übernommen.

Damit Saferpay als Popup geöffnet wird, muss die Funktion „OpenSaferpayTerminal()“ per <SCRIPT/> in die HTML Seite eingebunden werden:

```
<script SRC="http://www.saferpay.com/OpenSaferpayScript.js"></script>
```

5.3.1 FORM (POST)

Beispiel FORM: Saferpay öffnet sich im gleichen Fenster

zur Kasse...

Beispiel FORM: Saferpay öffnet sich im gleichen Fenster


```
<form method="post"
  action="https://www.saferpay.com/hosting/Redirect.asp">
  <input type="hidden" name="ACCOUNTID" value="99867-94913159" >
  <input type="hidden" name="AMOUNT" value="100" >
  <input type="hidden" name="CURRENCY" value="EUR" >
  <input type="hidden" name="DESCRIPTION" value="Saferpay Testeinkauf" >
  <input type="submit" name="SUBMIT" value="zur Kasse..." >
</form>
```

5.3.2 LINK (GET)

Beispiel LINK: Saferpay öffnet sich als Popup Fenster

[zur Kasse...](#)

Beispiel LINK: Saferpay öffnet sich im eigenen neuen Fenster


```
<TABLE borderColor=white borderColorDark=black bgColor=silver border=1>
<TR><TD>
  <a href="https://www.saferpay.com/hosting/Redirect.asp?_
    AMOUNT=100&ACCOUNTID=99867-94913159&CURRENCY=DEM&_
    DESCRIPTION=Saferpay Testeinkauf"
    onClick="OpenSaferpayTerminal(this.href, this, 'LINK')">
    zur Kasse...
  </a>
</TD></TR>
</TABLE>
```

5.3.3 BUTTON

Beispiel BUTTON: Saferpay öffnet sich als Popup Fenster

zur Kasse...

Beispiel BUTTON: Saferpay öffnet sich im eigenen neuen Fenster


```
<input type="BUTTON"
  onClick="OpenSaferpayTerminal('http://support.saferpay.de/scripts/
    Redirect.asp?AMOUNT=100&ACCOUNTID=99867-94913159&
    CURRENCY=DEM&DESCRIPTION=Testeinkauf Saferpay Gateway',
    this, 'BUTTON')"
  value="zur Kasse..." >
```