sage | pay

Sage Pay Token System
Integration and Protocol
Guidelines 3.00

Published: 01/08/2014

Table of Contents

Document Details

Version History

Date	Change	Page
01/11/2013	Document published.	---
01/05/2014	New screenshots.	---
	Included additional fields for Financial Institutions (MCC 6012).	25,42
	References to Sage Pay website updated.	---
	European Payment Information updated.	05
01/08/2014	Removed reference to Laser Cards and PayPal.	---
	Added PPro / PayPal indicators.	---

Legal Notice

This Protocol and Integration Guidelines document ("Manual") has been prepared to assist you with integrating your own (or your client's) service with Sage Pay's payment gateway. You are not permitted to use this Manual for any other purpose.

Whilst we have taken care in the preparation of this Manual, we make no representation or warranty (express or implied) and (to the fullest extent permitted by law) we accept no responsibility or liability as to the accuracy or completeness of the information contained within this Manual. Accordingly, we provide this Manual "as is" and so your use of the Manual is at your own risk.

In the unlikely event that you identify any errors, omissions or other inaccuracies within this Manual we would really appreciate it if you could please send details to us using the contact details on our website at www.sagepay.com.

We may update this Manual at any time without notice to you. Please ensure that you always use the latest version of the Manual, which we publish on our website at www.sagepay.com, when integrating with our payment gateway.

1.0 Introduction

This guide contains all essential information for the user to integrate Sage Pay's Token System using our Server or Direct integration methods. We recommend that you familiarise yourself with either of the above methods prior to implementing the Token System.

The accompanying Server and Direct Protocols can be found on sagepay.com.

Sage Pay's Token System is a process where card details are registered with the Sage Pay gateway, we'll securely store this information and return to you a unique identifier (`Token`) for you to store. This token can then be used to register future payments without the need to obtain the same card details again from your customer.

 Indicates additional information specific to European Payment method transactions.

 Indicates additional information specific to PayPal transactions.

2.0 Overview of Server Integration

There are two ways of registering a token:

2.1   Token Registration – Part of a Transaction

When registering a transaction with the Sage Pay gateway using either our Server or Direct integration methods, you can indicate as part of that post that you wish to create a `Token` for the card details being used in that transaction. The shopping process will still be as per the usual process where the customer will go through your website, select their chosen items and complete the checkout process. You will include the `CreateToken` field in your post and a token will be returned with a successful transaction response for you to store.

2.2   Token Registration – Standalone

If you wish to register a token without having to take a payment or send customer and shopping information, you can use our standalone token registration. This allows you to submit a POST to the Sage Pay gateway and simply exchange card details for a token. This means you can register card details for a token without your customer selecting any products, such as when your customer registers on your website or if they're adding/changing card details on the account they have on your website.

2.3   Using a Token

Once you've received a token from Sage Pay and this has been stored in your database against the customer details, you can use the token as a substitute for card data whenever that customer makes a purchase on your website.
Your server would simply submit the token with the normal transaction registration POST to the Sage Pay gateway. Once received, Sage Pay will validate the information sent and use the token provided to locate the card details from the Sage Pay database to send onto the bank for authorisation.

Please note that the Sage Pay Token System does not validate any customer information associated with the token and therefore it is important for you to store the correct token with the customer's details to ensure that payment is taken from the correct card.

A response regarding the outcome of the authorisation will be returned to you along with all the usual checks.
It is important to note that when you use a `Token`, you will need to indicate whether you wish for the token to be stored for use again using the `StoreToken` field.

2.4   Removing a Token

Tokens will automatically be removed when the card expiry date has passed. They will also be removed if you do not indicate that you wish to store the token after each use.
Alternatively, you can submit a request to the Sage Pay gateway to delete a token, such as when your customer deletes their account on your website or wishes to replace the stored card with another.

It is not possible to generate a `Token` using a European Payment method.

3.0 Registering a Token

3.1   Server Integration – Part of a Transaction
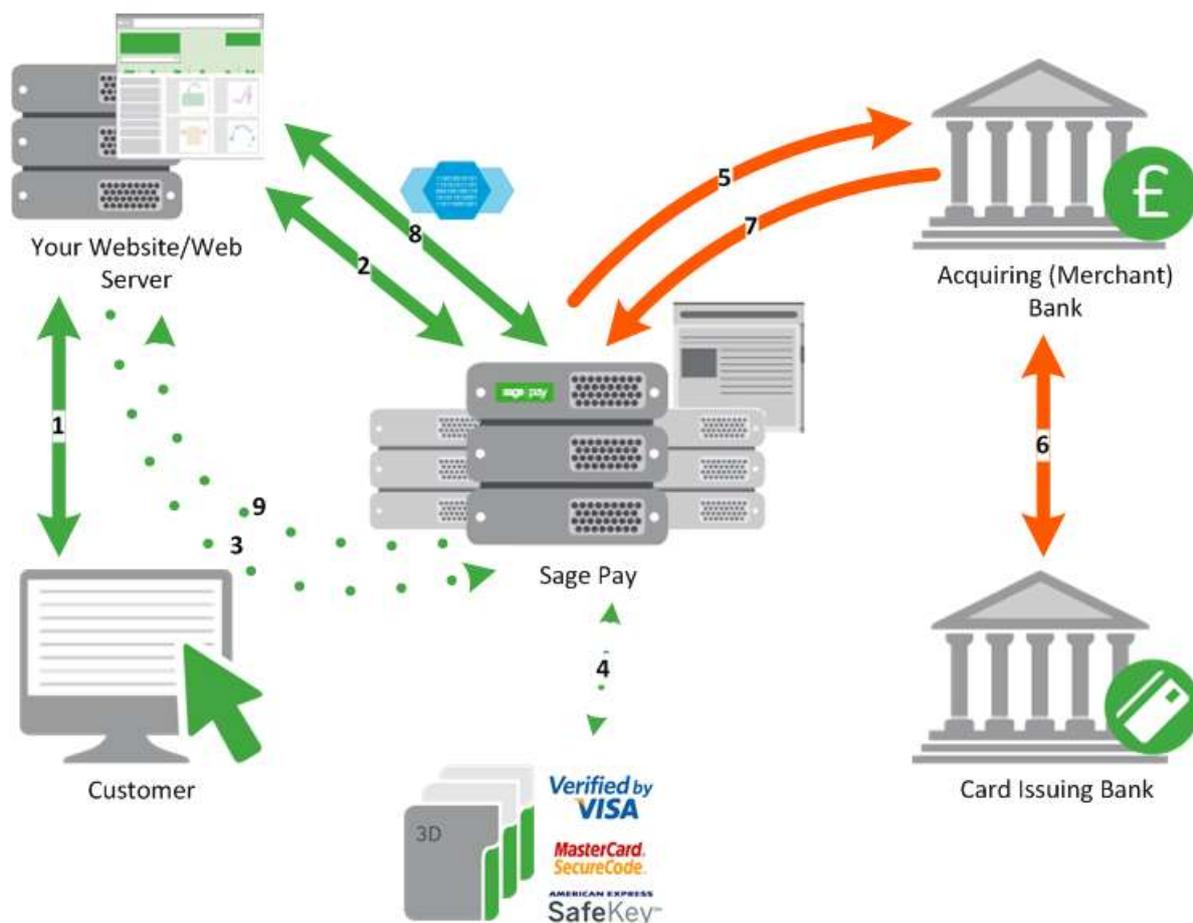
For a more detailed transaction flow, please refer to the Server Integration and Protocol Guidelines 3.00.

1. The customer visits your website and selects the products they wish to purchase.

2. When the customer is ready to pay for the goods, your server makes a POST to the Sage Pay gateway using the Server Protocol 3.00 with the `TxType`=**PAYMENT**, **DEFERRED** or **AUTHENTICATE** and `CreateToken`=**1**.

    If the POST is valid, Sage Pay will return a response and pass a `NextURL` back to you.

3. You will then be able to redirect your customer to the Sage Pay hosted payment pages using the `NextURL` provided.

    Once on the payment page your customer will enter their card details.

    Sage Pay will validate the entry of these details. As per the current protocol, Sage Pay will allow for 3 attempts. 3 failed attempts to register the card details will result in a failure notification being sent back to your server.

4. If the details the customer enters are valid as per current card checks and validation rules, the customer will continue through the 3D-Secure process (if enabled).

5. Card details will be sent to your acquiring back for authorisation.

6. Your acquiring bank gains authorisation from the card issuer.

7. The acquiring bank returns the result of the authorisation to Sage Pay.

8. If authorisation is successful then Sage Pay will store the CardNumber, ExpiryDate, CardHolder, CardType and generate a `Token`.

   Sage Pay will then send a Notification POST to your NotificationURL with the results of the transaction (the NotificationURL is the callback URL provided in your transaction registration POST). Sage Pay will include the `Token` in this POST to your server.

   Your server will then need to respond to the Notification Post with a `Status` and a `RedirectURL`. If the `Status` is anything other than **OK**, Sage Pay will fail the transaction and delete the `Token`.

9. Once a Notification response has been received, Sage Pay will redirect the customer to the `RedirectURL` provided.

3.2 Server Integration – Standalone
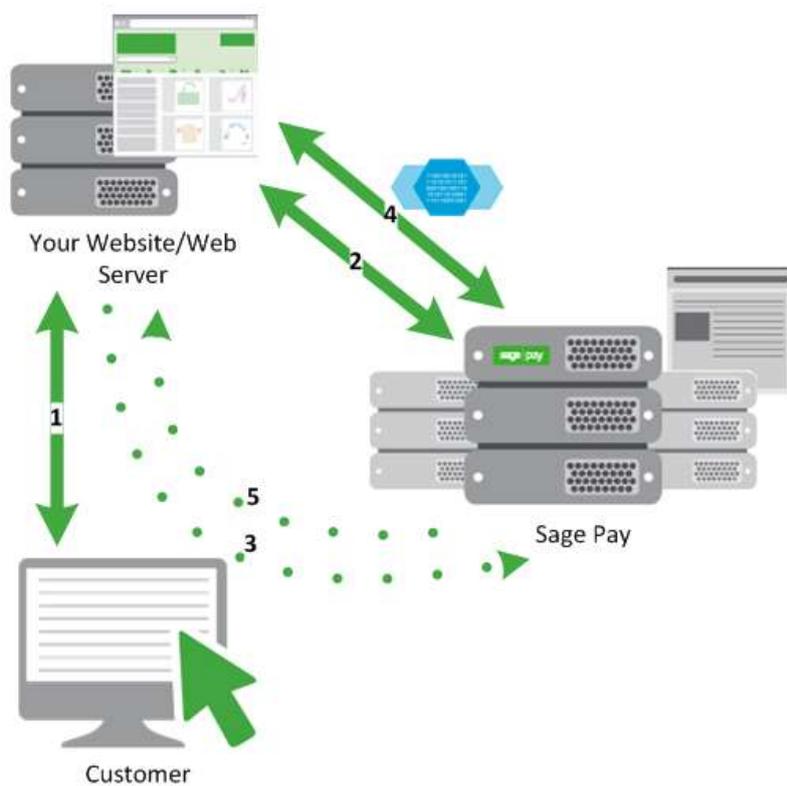
1. The customer visits your website and indicates they wish to store their card details for future use.

2. When you are ready to register for a token, your server makes a POST to the Sage Pay gateway using the Token System Protocol with the `TxType` **TOKEN**.

If the POST is valid, Sage Pay will return a response and pass a `NextURL` back to you.

3. You will then be able to redirect your customer to the Sage Pay hosted payment pages using the `NextURL` provided.

Sage Pay will validate the entry of these details. As per the current protocol, Sage Pay will allow for 3 attempts. 3 failed attempts to register the card details will result in a failure notification being sent back to your server.

If the details the customer enters are valid as per current card checks and validation rules, Sage Pay will store the CardNumber, ExpiryDate, CardHolder, CardType, CV2 (if supplied) and generate a `Token`.

4. Sage Pay will then send a Notification POST to your `NotificationURL` with the results of the transaction (the `NotificationURL` is the callback URL provided in your transaction registration POST). Sage Pay will include the `Token` in this POST to your server.

Your server will then need to respond to the Notification Post with a `Status` and a `RedirectURL`. If the `Status` is anything other than OK, Sage Pay will fail the transaction and delete the `Token` and associated card details.

5. Once a Notification response has been received, Sage Pay will redirect the customer to the `RedirectURL` provided.

The CV2 value is only stored prior to authorisation. After the initial attempted use of the token the CV2 will be deleted to comply with industry regulations. If you have AVS/CV2 checks enabled on your account, you will need to obtain this information from the customer each time the token is used.

3.3   Direct Integration – Part of a Transaction
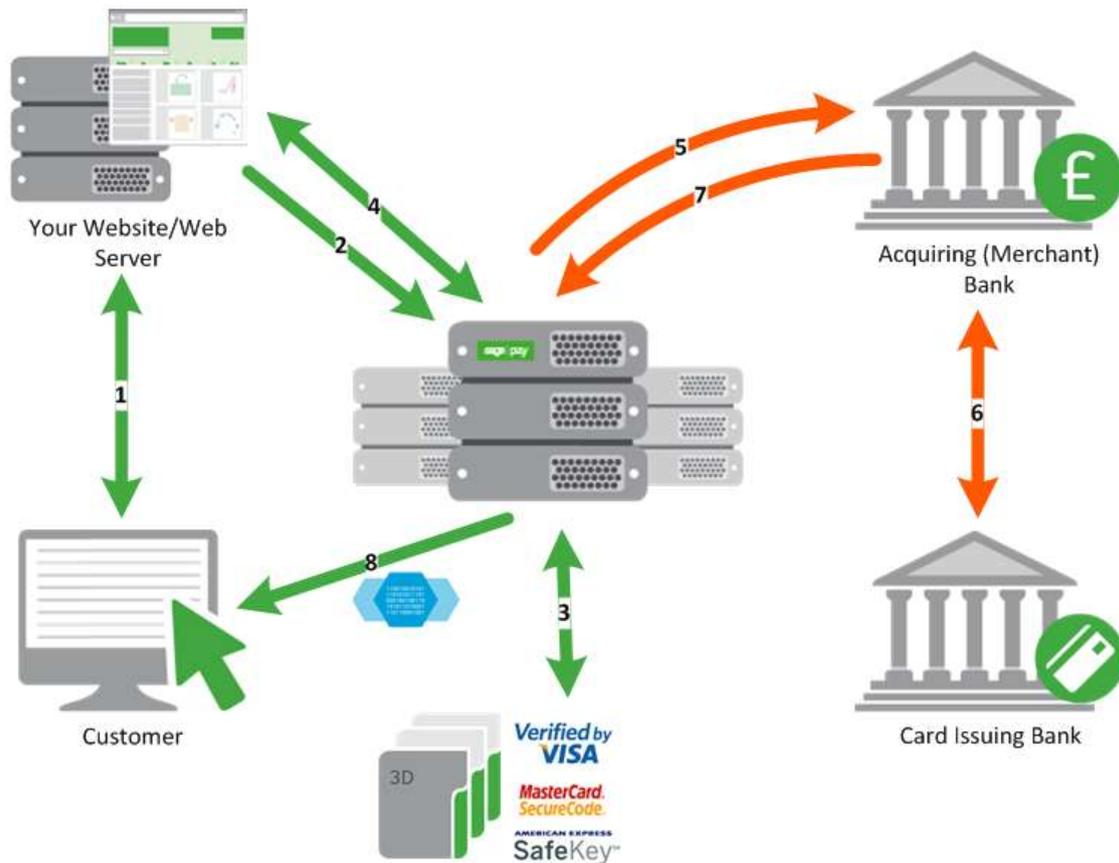
For a more detailed transaction flow, please refer to the Direct Integration and Protocol Guidelines 3.00.

1. The customer visits your website and selects the products they wish to purchase and enters their payment details directly on your website.

2. When the customer is ready to pay for the goods, your server makes a POST to the Sage Pay gateway using the Direct Protocol 3.00 with the `TxType` =**PAYMENT**, **DEFERRED** or **AUTHENTICATE** and `CreateToken`=**1**.

    Sage Pay will validate the POST. If it is invalid, your server will be notified and we recommend that you check your submission and rePOST.

3. If the POST is valid, Sage Pay will verify 3D-Secure enrolment.

4. Sage Pay replies to your POST with the necessary details to complete 3D-Secure. After 3D-Secure is complete, you POST the results to Sage Pay.

5. Card details will be sent to your acquiring back for authorisation.

6. Your acquiring bank gains authorisation from the card issuer.

7. The acquiring bank returns the result of the authorisation to Sage Pay.

8. Sage Pay will then send a response to your server with the result of the transaction including the `Token`.

3.4 Direct Integration - Standalone
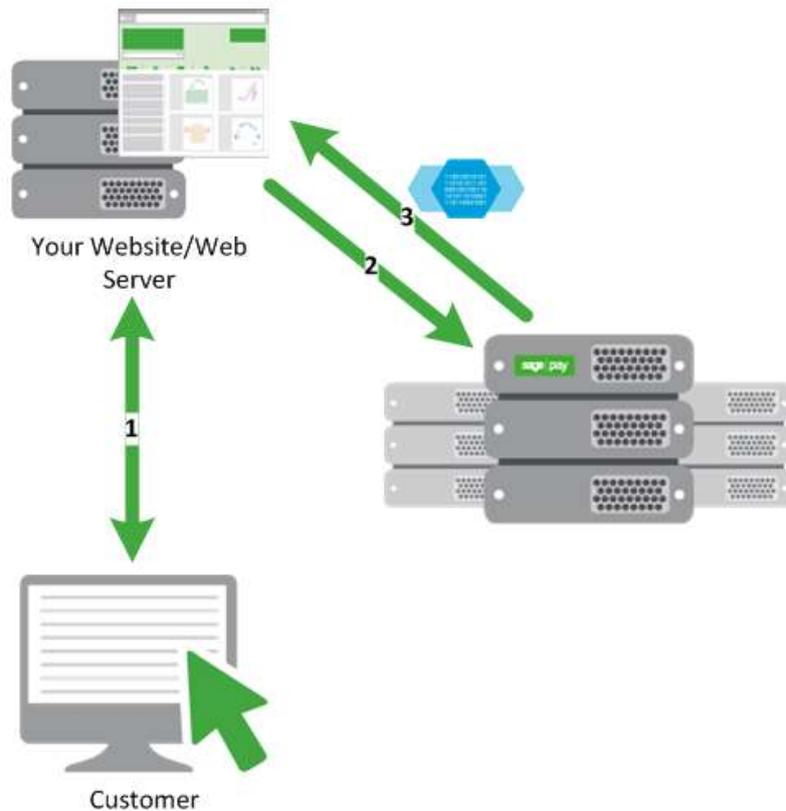
1. The customer visits your website and indicates they wish to store their card details for future use.
   The customer enters their card details directly on your website.

2. Your server makes a POST to the Sage Pay gateway using the Token System protocol with the `TxType`=**TOKEN**.

   Sage Pay will validate the POST. If it is invalid, your server will be notified and we recommend that you check your submission and repost.

   If the POST is valid as per current card checks and validation rules, Sage Pay will store the CardNumber, ExpiryDate, CardHolder, CardType, CV2 (if supplied) and generate a token.

3. Sage Pay will send the result of the token registration back to your server with the `Token`.

> The CV2 value is only stored prior to authorisation. After the initial attempted use of the token the CV2 will be deleted to comply with industry regulations. If you have AVS/CV2 checks enabled on your account, you will need to obtain this information from the customer each time the token is used.

4.0 Using a Token

4.1   Server Integration
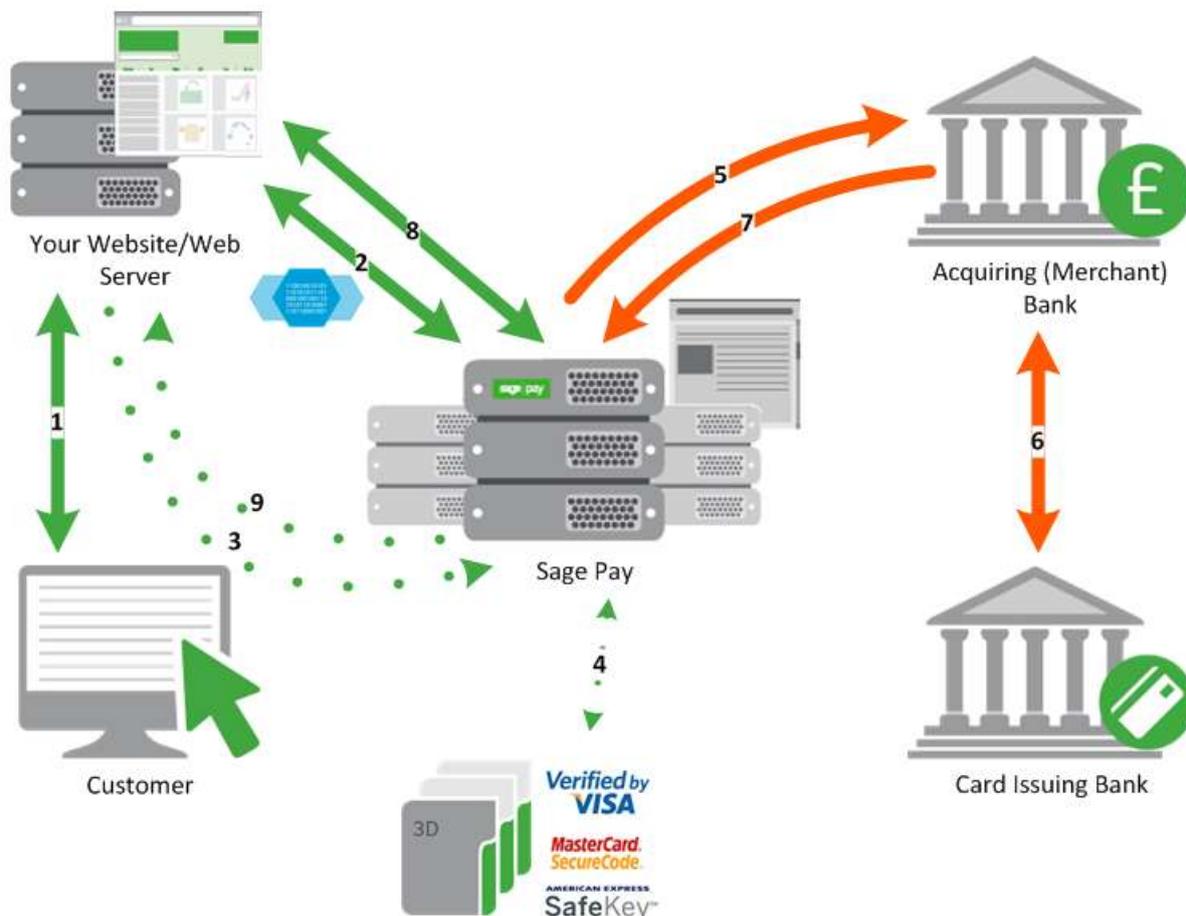
For a more detailed transaction flow, please refer to the Server Integration and Protocol Guidelines 3.00.

1. Returning customer visits your website where you are able to associate them with a pre-registered `Token`.

2. Your server creates and submits a standard transaction registration POST including the `Token` and `TxType` =**PAYMENT**, **DEFERRED** or **AUTHENTICATE** as well as all the other relevant protocol values. This is POSTed to the usual Sage Pay Server transaction registration payment URL.

   Sage Pay will validate the `Token` against your account. If the `Token` is invalid, your server will be notified and we would recommend that you check the token and rePOST.

   If the `Token` is valid, Sage Pay will validate the rest of the POST, as per the normal Server Integration and Protocol Guidelines. If the POST is invalid, your server will be notified and you will have to check the information sent and rePOST.

   If the POST is valid, Sage Pay will pass your server the `NextURL`.

3. At this point, Sage Pay will obtain the card details associated with the `Token`.

If a CV2 value is available, Sage Pay will pull this from the token database. If no CV2 value is available and it is required on your account, then the shopper will be given the opportunity to re-enter their CV2 number on the payment page, which will be an adjusted version of the standard card details page, with only a CV2 entry box. If no CV2 is entered, none exists in our database and the account in question requires a CV2 check, then the transaction will be rejected and your server will be notified.

4. If a CV2 is available in the database or the customer is given the opportunity to enter the CV2 and does so then the customer will continue through the 3D-Secure process (if enabled).

5. Card details will be sent to your acquiring back for authorisation.

6. Your acquiring bank gains authorisation from the card issuer.

7. The acquiring bank returns the result of the authorisation to Sage Pay.

8. If bank authorisation is successful and passes the AVS/CV2 checks on your account (if present) then the transaction is authorised and a successful notification is sent to your server.

    If bank authorisation fails, or is authorised but failed by the AVS/CV2 rulebase then the transaction is declined, or rejected, and a failure notification is sent to your server.

    Once a transaction has completed as either Failed or Authorised then the token is considered as used and will be deleted. To store the token a request must be sent as per the transaction registration post `StoreToken`=**1**. If the `StoreToken` field is not passed or where `StoreToken`=**0** the token will be deleted by default.

4.2   Direct Integration
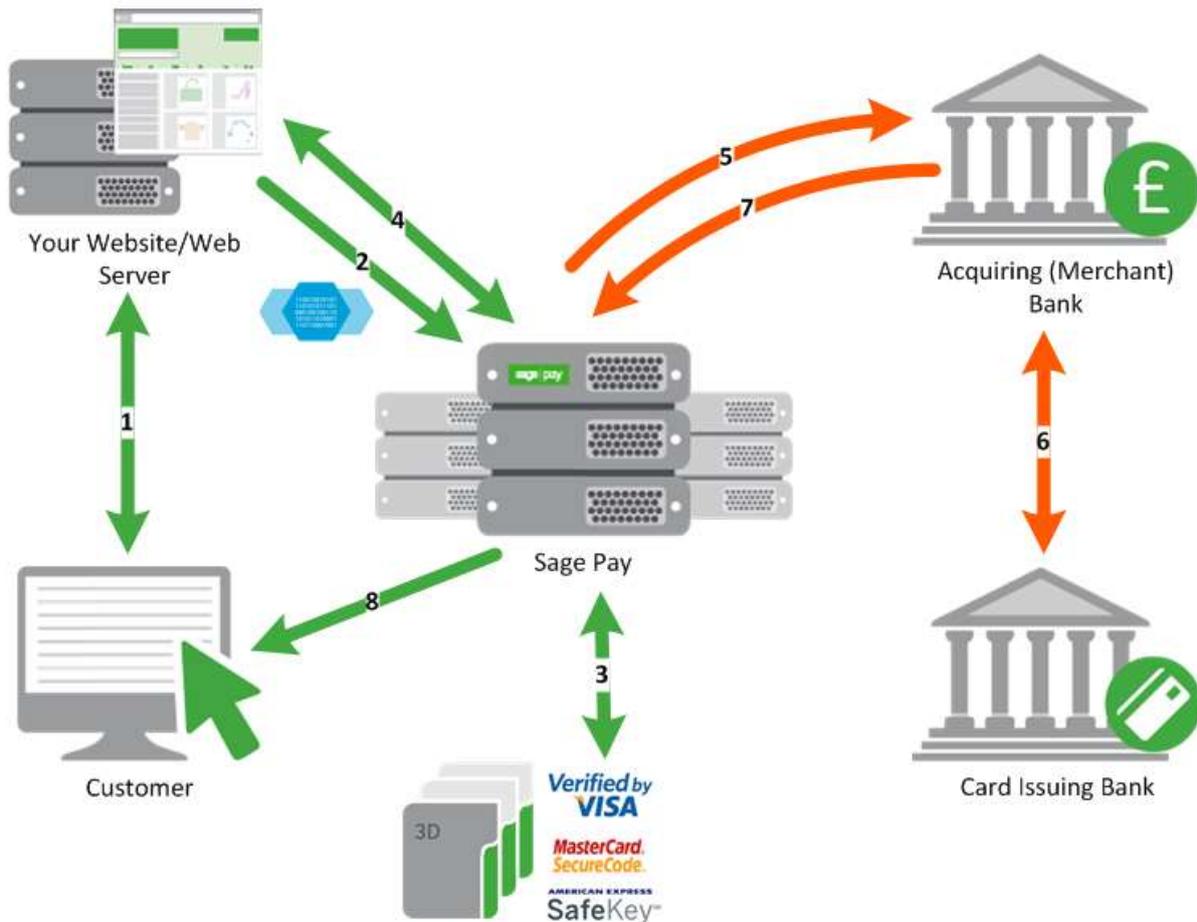
For a more detailed transaction flow, please refer to the Direct Integration and Protocol Guidelines 3.00

1. Returning customer visits your website where you are able to associate them with a pre-registered `Token`.

2. Your server creates and submits a standard transaction registration POST including the `Token` and `TxType` =**PAYMENT**, **DEFERRED** or **AUTHENTICATE** as well as all the other relevant protocol values. This is POSTed to the usual Sage Pay Direct transaction registration payment URL.

    Sage Pay will validate the `Token` against your account. If the `Token` is invalid, your server will be notified and we would recommend that you check the token and rePOST.

    If the `Token` is valid, Sage Pay will validate the rest of the POST, as per the normal Server Integration and Protocol Guidelines. If the POST is invalid, your server will be notified and you will have to check the information sent and rePOST.

    If a `CV2` value (optional) is passed within the POST then this value will always be used in place of any `CV2` value already stored within the Sage Pay Token database.

    If a `CV2` value is not passed within the POST then Sage Pay will check to see if the `Token` has been used before. If the `Token` has not been used before, a `CV2` value will be pulled from the `Token` database.

If the CV2 value has been used before and a CV2 value has not been supplied in the initial POST, Sage Pay check to see if a CV2 is required for the transaction (i.e. for AVS/CV2 checks). If a CV2 value is required but has not been supplied, the transaction will be **REJECTED** and your server will be notified.

3. If a CV2 value is supplied in the initial POST or a CV2 is obtained from the Sage Pay Token database then the card details are forwarded to the 3D-Secure directory.
   If no CV2 value is supplied, nor is one available on record and the transaction does not require a CV2 (i.e. no AVS/CV2 checks) then the transaction is also forwarded to the 3D-Secure directory.

4. Sage Pay replies to your POST with the necessary details to complete 3D-Secure. After 3D-Secure is complete, you POST the results to Sage Pay.

5. Card details will be sent to your acquiring back for authorisation.

6. Your acquiring bank gains authorisation from the card issuer.

7. The acquiring bank returns the result of the authorisation to Sage Pay.

8. If the transaction is authorised by the bank, then Sage Pay will check your AVS/CV2 rulebase. If the checks pass your rulebase then the transaction is authorised and a successful response is sent back to your server.

   If the transaction is authorised by the bank and after Sage Pay checks your AVS/CV2 rulebase, the transaction fails to pass this rulebase, then a rejected response is sent back to your server. If the bank declines the authorisation then a failed response is sent.

   Once a transaction has completed as either successful or failed, the token is considered as used. For successful transactions the token is deleted unless a request is made as per the transaction registration post StoreToken=**1**.For failed transactions the token will continue to be stored unless subsequent attempts are successful or a **REMOVETOKEN** request is made.

It is your responsibility to remove tokens for failed transactions by using the **RemoveToken** request with TxType=**REMOVETOKEN**.

5.0 Removing a Token

5.1 Server and Direct Integration
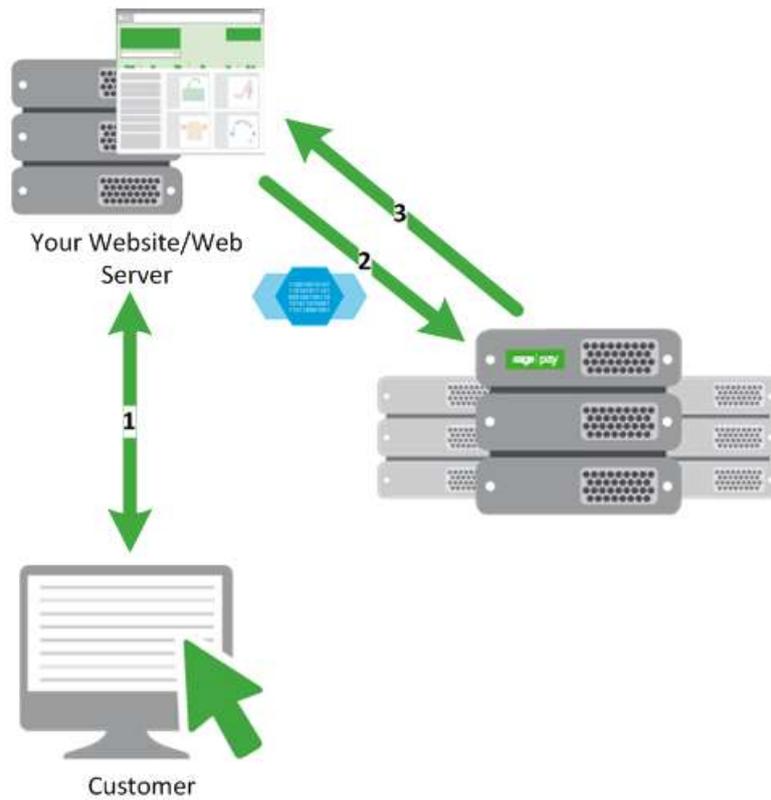
1. You want to remove a token, or a returning customer removes their card details from their account on your website.

2. Your server creates and submits a POST with the `TxType` =**REMOVETOKEN** along with the other required protocol values. This is POSTed to the remove token URL. Whether you use Server or Direct for payment registration, the **REMOVETOKEN** request is sent to the same URL and the requests and responses do not differ.

   Sage Pay will validate the POST then validate the `Token` against your account. If either the POST or the `Token` are invalid, your server will be notified and we would recommend that you check and rePOST.

3. If the POST and the `Token` are valid, Sage Pay will respond to your server with the result of your **REMOVETOKEN** request and where applicable, delete the `Token` from our database.

6.0 Character Sets and Encoding

All transactions are simple synchronous HTTPS POSTs sent from a script on your servers to the Sage Pay gateway, with the same script reading the Response component of that POST to determine success or failure. These POSTs can be sent using any HTTPS compatible objects (such as cURL in PHP, HttpWebRequest in .NET and Apache HttpComponents in Java).

The data should be sent as URL Encoded Name=Value pairs separated with & characters and sent to the Sage Pay Server URL with a Service name set to the message type in question.

The following sections detail the contents of the POSTs and responses, between your server and ours. The format and size of each field is given, along with accepted values and characters. The legend below explains the symbols:

Aa	Letters (A-Z and a-z)	^	Caret	+	Plus
0-9	Numbers	[]	Square brackets	()	Parentheses
á	Accented characters	*	Asterisk	;	Semi-colon
&	Ampersand	'	Apostrophe (single quote)	|	Pipe
@	At sign	/\	Slash and Backslash	!	Exclamation Mark
:	Colon	-	Hyphen		Space
,	Comma	_	Underscore	~	Tilde
{}	Curly brackets	.	Full stop / Period	=	Equals
"	Quotes	$	Dollar	US	Valid 2-letter US States
#	Hash	?	Question Mark	DATE	Date in the format YYYY-MM-DD
ISO639	ISO 639-2 (2-letter language codes)	BASE64	Valid Base64 characters (A-Z,a-z,0-9,+ and /)	BOOLEAN	True or False
ISO3166	ISO 3166-1 (2-letter country codes)	CR / LF	New line (Carriage Return and Line Feed)	RFC532N	RFC 5321/5322 (see also RFC 3696) compliant email addresses
ISO4217	ISO 4217 (3-letter currency codes)	RFC1738	RFC 1738 compliant HTTP(S) URL All non-compliant characters, including spaces should be URL encoded	<HTML>	Valid HTML with no active content. Script will be filtered. Includes all valid letters, numbers, punctuation and accented characters

Appendix A: Part of a Transaction Token Registration – SERVER integration

A1.   You submit your transaction registration POST

This is performed via a HTTPS POST request, sent to the initial Sage Pay Payment URL service vspserver-register.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 .	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**PAYMENT** **DEFERRED** **AUTHENTICATE**	See companion document "Server Integration and Protocol Guidelines 3.00" for more information on the different transaction types. The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
VendorTxCode	Yes	Aa 0-9 {} . _ -	40 chars		This should be your own reference code to the transaction. Your site should provide a completely unique VendorTxCode for each transaction.
Amount	Yes	0-9 . ,		0.01 to 100,000.00	Amount for the transaction containing minor digits formatted to 2 decimal places where appropriate. e.g. 5.10 or 3.29. Values such as 3.235 will be rejected. Minimum for no minor unit currencies like JPY is 1. Amounts must be in the UK currency format. The period must be used to indicate the decimal place. The comma must only be used to separate groups of thousands.
Currency	Yes	ISO4217	3 chars	ISO 4217 Examples: **GBP**, **EUR** and **USD**	The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.

Description	Yes	`<HTML>`	100 chars		Free text description of goods or services being purchased. This will be displayed on the Sage Pay Server payment page as the customer enters their card details.
NotificationURL	Yes	RFC1738	255 chars		This should be the fully qualified URL (including http:// or https:// header). It is the callback URL to which Notification POSTs are sent.
BillingSurname	Yes	Aa á /\ & - - ' , 0-9	20 chars		Customer billing details. All mandatory fields must contain a value, apart from the `BillingPostcode`. The `BillingPostcode` can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error. The `BillingState` becomes mandatory when the `BillingCountry` is set to **US**.
BillingFirstnames	Yes	Aa á /\ & - - ' , 0-9	20 chars		
BillingAddress1	Yes	Aa á /\ & - - ' , 0-9 : + () CR/LF	100 chars		
BillingAddress2	No	Aa á /\ & - - ' , 0-9 : + () CR/LF	100 chars		
BillingCity	Yes	Aa á /\ & - - ' , 0-9 : + () CR/LF	40 chars		
BillingPostCode	Yes	Aa - 0-9	10 chars		
BillingCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
BillingState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
BillingPhone	No	0-9 - Aa + ()	20 chars		
DeliverySurname	Yes	Aa á /\ & - - ' , 0-9	20 chars		Customer delivery details. All mandatory fields must contain a value, apart from the `DeliveryPostcode`. The `DeliveryPostcode` can be blank for countries that do not have postcodes
DeliveryFirstnames	Yes	Aa á /\ & - - ' , 0-9	20 chars		

DeliveryAddress1	Yes	Aa á /\ & - _ ' , 0-9 : + () CR/LF	100 chars		(e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error.
DeliveryAddress2	No	Aa á /\ & - _ ' , 0-9 : + () CR/LF	100 chars		
DeliveryCity	Yes	Aa á /\ & - _ ' , 0-9 : + () CR/LF	40 chars		The DeliveryState becomes mandatory when the DeliveryCountry is set to US.
DeliveryPostCode	Yes	Aa - 0-9	10 chars		
DeliveryCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
DeliveryState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
DeliveryPhone	No	0-9 - Aa + ()	20 chars		
CustomerEMail	No	RFC532N	255 chars	Examples: me@mail1.com:me@mail2.com	The customers email address. If you wish to use multiple email addresses, you should add them using the : (colon) character as a separator. The current version of the Server integration method does not send confirmation emails to the customer. This field is provided for your records only.
Basket	No	<HTML>	7500 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the Basket field.	You can use this field to supply details of the customer's order. This information will be displayed to you in "My Sage Pay". If this field is supplied then the BasketXML field should not be supplied.
AllowGiftAid	No	BOOLEAN	Flag	**0** (default) **1**	This flag allows the gift aid acceptance box to appear for this transaction on the payment page. This only appears if your vendor account is Gift Aid enabled. **0 =** No Gift Aid box displayed (default) **1 =** Display Gift Aid box on payment page.

ApplyAVSCV2	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the AVS/CV2 checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If AVS/CV2 enabled then check them. If rules apply, use rules (default) **1 =** Force AVS/CV2 checks even if not enabled for the account. If rules apply, use rules. **2 =** Force NO AVS/CV2 checks even if enabled on account. **3 =** Force AVS/CV2 checks even if not enabled for the account but DON'T apply any rules.
Apply3DSecure	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the 3D Secure checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If 3D-Secure checks are possible and rules allow, perform the checks and apply the authorisation rules. (default) **1 =** Force 3D-Secure checks for this transaction if possible and apply rules for authorisation. **2 =** Do not perform 3D-Secure checks for this transaction and always authorise. **3 =** Force 3D-Secure checks for this transaction if possible but ALWAYS obtain an auth code, irrespective of rule base.

Profile	No	Aa	10 chars	**NORMAL** (default) **LOW**	A profile of **LOW** returns the simplified payment pages which have only one step and minimal formatting. Designed to run in i-Frames. Omitting this field or sending **NORMAL** renders the normal card selection screen.
AccountType	No	Aa	1 char	**E** (default) **M** **C**	This optional flag is used to tell the Sage Pay gateway which merchant account to use. If omitted, the system will use E, then M, then C by default. **E =** Use the e-commerce merchant account (default). **M =** Use the mail order/telephone order account (if present). **C =** Use the continuous authority merchant account (if present).
CreateToken	No	0-9	Flag	**0** (default) **1**	Use this flag to indicate you wish to have a token generated and stored in our database and returned to you for future use. **0 =** This will not create a token from the payment (default) **1 =** This will create a token from the payment if successful and return a `Token`.
BasketXML	No		20000 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the `BasketXML` field.	A more flexible version of the current basket field which can be used instead of the basket field. If this field is supplied then the Basket field should not be supplied.
CustomerXML	No		2000 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the `CustomerXML` field.	This can be used to supply information on the customer for purposes such as fraud screening.
SurchargeXML	No		800 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the `SurchargeXML` field.	Use this field to override current surcharge settings in "My Sage Pay" for the current transaction. Percentage and fixed amount surcharges can be set for different payment types.

VendorData	No	Aa 0-9	200 chars		Use this field to pass any data you wish to be displayed against the transaction in "My Sage Pay".
ReferrerID	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 char		This can be used to send the unique reference for the Partner that referred the Vendor to Sage Pay.
Language	No	ISO639	2 chars	ISO 639-2 Examples: **EN**, **DE** and **FR**	The language the customer sees the payment pages in is determined by the code sent here. If this is not supplied then the language default of the shoppers browser will be used. If the language is not supported then the language supported in the templates will be used. Currently supported languages in the Default templates are: French, German, Spanish, Portuguese, Dutch and English.
Website	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		Reference to the website this transaction came from. This field is useful if transactions can originate from more than one website. Supplying this information will enable reporting to be performed by website.
FIRecipientAcctNumber	No	Aa 0-9	10 chars		This should either be the first 6 and the last 4 characters of the primary recipient PAN (no spaces). Where the primary recipient account is not a card this will contain up to 10 characters of the account number (alphanumeric), unless the account number is less than 10 characters long in which case the account number will be present in its entirety. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientSurname	No	Aa	20 chars		This is the surname of the primary recipient. No special characters such as apostrophes or hyphens are permitted. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)

| FIRecipientPostcode | No | Aa 0-9 | | | This is the postcode of the primary recipient. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions) |
| FIRecipientDoB | No | 0-9 | | | This is the date of birth of the primary recipient in the format YYYYMMDD This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions) |

Please refer to the Sage Pay Server Integration and Protocol Guidelines 3.00 for the response to the above POST and subsequent POSTs required to complete the transaction process.

Appendix B: Standalone Token Registration – SERVER integration

B1.   You submit your token registration POST

This is performed via a HTTPS POST request, sent to the Sage Pay token registration URL service token.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 .	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**TOKEN**	The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
VendorTxCode	Yes	Aa 0-9 {} . _ -			This should be your own reference code to the transaction. Your site should provide a completely unique VendorTxCode for each transaction.
Currency	Yes	ISO4217			The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.
NotificationURL	Yes	RFC1738			This should be the fully qualified URL (including http:// or https:// header). It is the callback URL to which Notification POSTs are sent (see B3)
Profile	No	Aa		**NORMAL** (default) **LOW**	A Profile of **LOW** returns the simplified payment pages which have only one step and minimal formatting. Designed to run in i-Frames. Omitting this field or sending **NORMAL** renders the normal card selection screen.

Language	No	ISO639		ISO 639-2 Examples: **EN**, **DE** and **FR**	The language the customer sees the payment pages in is determined by the code sent here. If this is NULL then the language default of the shoppers browser will be used. If the language is not supported then the language supported in the templates will be used. Currently supported languages in the Default templates are: French, German, Spanish, Portuguese, Dutch and English.

B2. Server response to the token registration POST

This is the plain text response part of the POST originated by your servers in B1. Encoding will be as Name=Value pairs separated by carriage return and linefeeds (CRLF).

Respone format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	`0-9` `.`	4 chars	**3.00**	Protocol version used by the system. Same as supplied in B1.
Status	Yes	`Aa` ▪	15 chars	**OK** **OK REPEATED** **MALFORMED** **INVALID** **ERROR**	If the `Status` is not **OK**, the `StatusDetail` field will give more information about the problem. **OK =** Process executed without error. **OK REPEATED =** If the `VendorTxCode` passed in B1 has been used before, but that transaction is still active, then details of that transaction are passed back in this POST and the suffix **REPEATED** is appended to the `Status`. Your system must be able to handle **REPEATED** messages from Sage Pay. **MALFORMED =** Input message was missing fields or badly formatted – normally will only occur during development. **INVALID =** Transaction was not registered because although the POST format was valid, some information supplied was invalid. E.g. incorrect vendor name or currency. **ERROR =** A problem occurred at Sage Pay which prevented transaction registration. Please notify Sage Pay if a `Status` of **ERROR** is seen, together with your `Vendor`, `VendorTxCode` and the `StatusDetail`.

StatusDetail	Yes	Aa 0-9 - ▢ () , :	255 chars		Human-readable text providing extra detail for the `Status` message. Always check `StatusDetail` if the `Status` is not **OK**
VPSTxId	Yes	Aa 0-9 - {}	38 chars		The Sage Pay ID to uniquely identify the transaction on our system. Only present if `Status` is **OK** or **OK REPEATED**.
SecurityKey	Yes	Aa 0-9	10 chars		A Security key which Sage Pay uses to generate a MD5 Hash for to sign the Notification message (B3 below). The signature is called `VPSSignature`. This value is used to allow detection of tampering with notifications from the Sage Pay gateway. It must be kept secret from the customer and held in your database. Only present if `Status` is **OK** or **OK REPEATED**.
NextURL	Yes	RFC1738	255 chars		This is the URL to which the Vendor must redirect the Customer to continue the transaction. Only present if `Status` is **OK** or **OK REPEATED**. Note that the full URL must be used for the redirect, including any appended parameters.

B3. Notification of Results for Token

The Sage Pay Server will send notification in the request part of a POST to the Notification URL provided in B1. The request will be URL encoded, with Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Returned Values	Description
VPSProtocol	Yes	0-9 .	4 chars	**3.00**	Protocol version used by the system. Same as supplied in B1.
TxType	Yes	Aa	15 chars	**TOKEN**	Same as supplied in B1.
VendorTxCode	Yes	Aa 0-9 {} . _ −	40 chars		Same as supplied in B1.
Status	Yes	Aa	15 chars	**OK** **ABORT** **REJECTED** **ERROR**	Result of the token registration. If the Status is not **OK**, the StatusDetail field will give more information about the problem. **OK =** The card registration has been successful. **ABORT =** The card registration could not be completed because the user clicked the CANCEL button on the payment pages, or went inactive for 15 minutes or longer. **REJECTED =** The Sage Pay System rejected the card registration because the registration attempts has been exceeded. **ERROR =** An error occurred at Sage Pay which meant the transaction could not be completed successfully.
VPSTxId	Yes	Aa 0-9 .	38 chars		Same as returned in B2. Curly brackets { } are NOT returned with the VPSTxId in this instance and should NOT be included for the VPSTxId when constructing your MD5.

CardType	No	`Aa`	15 chars	**VISA** **MC** **MCDEBIT** **DELTA** **MAESTRO** **UKE** **AMEX** **DC** **JCB**	**VISA** is Visa **MC** is MasterCard **MCDEBIT** is Debit MasterCard **DELTA** is Visa Debit **MAESTRO** is Domestic and International issued Maestro **UKE** is Visa Electron **AMEX** is American Express **DC** is Diners Club International and Discover **JCB** is Japan Credit Bureau
Token	No	`Aa` `0-9` `-` `{}`	38 chars		The token generated by Sage Pay.
StatusDetail	Yes	`Aa` `0-9` `-` `_` `()` `,` `:`	255 chars		Human-readable text providing extra detail for the `Status` message. Always check `StatusDetail` if the `Status` is not **OK**
Last4Digits	No	`0-9`	4 chars		The last 4 digits of the card number used in this transaction. This field is supplied to allow merchants using wallet and token systems to identify the card to their customers.
VPSSignature	Yes	`Aa` `0-9`	100 chars	MD5 signature of the concatenation of the values of: `VPSTxId` + `VendorTxCode` + `Status` + `VendorName` + `{ Token}` + `SecurityKey`	To detect any possible tampering with messages, your site should compute the same MD5 signature (which incorporates the `SecurityKey` provided in B2) and check it against `VPSSignature`. You can then decide what to do with transactions that appear to have been tampered with. MD5 value is returned in UPPER CASE. If a field is returned without a value this should not be included in the string. Please ensure the `VendorName` is lower case prior to hashing.
ExpiryDate	No	`0-9`	4 chars		Expiry date of card used in the format of **MMYY**

B4.   You acknowledge receipt of Notification POST

This is the plain text response part of the POST originated by the Server in the step above. Encoding must be as Name=Value fields separated by carriage-return-linefeeds (CRLF).

Response format

Name	Mandatory	Format	Max Length	Returned Values	Description
Status	Yes	Aa 0-9	20 chars	**OK** **INVALID** **ERROR**	**OK =** Send this if you successfully received the notification post in B3. **INVALID =** Send this if the details you received in the B3 post were not consistent with expectations for this Transaction. The RedirectURL must still be provided, and Sage Pay will still redirect the customer back to your site. Only send this result if you want to cancel the transaction. **ERROR =** An error has occurred during your Notification processing. The Sage Pay system will check for a RedirectURL, and if one is provided the customer will be redirected to your site. Only send this result if you want to cancel the transaction and report an **ERROR** to Sage Pay. **INVALID** or **ERROR** Status will prevent the transaction from settling, so the customer will not be charged. You should send **OK** in all circumstances where no errors occur in validating the Notification POST, so even if Sage Pay send you a Status of **ABORT** or **NOTAUTHED** in B3 above, you should reply with an **OK** Status and a RedirectURL that points to a page informing the customer that the transaction did not complete.

RedirectURL	No	RFC1738	255 chars		This should be the fully qualified URL (including http:// or https:// header). It is the callback URL to which Notification POSTs are sent (see B3) If you wish to pass parameters back to your own site (such as the session id or transaction code), these should be included in RedirectURL.
StatusDetail	No	Aa 0-9 . ( ) , :	255 chars		Human-readable text providing extra detail for the Status message. If Status is not **OK** state what is wrong with the transaction and why you are rejecting it.

IMPORANT NOTE: Before writing the three fields above to the Response object of the POST, please ensure you clear your response buffer to remove any header code, comments or HTML. The Sage Pay Server is expecting "Status =" to be the first characters in the response. If it does not see these, it treats the response as though it is an error and fails the transaction! Also, all POSTs must be communicated through ports 80 and 443.

Appendix C: Part of a Transaction Token Registration – DIRECT integration

C1.   You submit your transaction registration POST

This is performed via a HTTPS POST request, sent to the initial Sage Pay Payment URL service vspdirect-register.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 .	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**PAYMENT** **DEFERRED** **AUTHENTICATE**	See companion document "Direct Integration and Protocol Guidelines 3.00" for more information on the different transaction types. The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
VendorTxCode	Yes	Aa 0-9 {} . _ -	40 chars		This should be your own reference code to the transaction. Your site should provide a completely unique VendorTxCode for each transaction.
Amount	Yes	0-9 . ,		0.01 to 100,000.00	Amount for the transaction containing minor digits formatted to 2 decimal places where appropriate. e.g. 5.10 or 3.29. Values such as 3.235 will be rejected. Minimum for no minor unit currencies like JPY is 1. Amounts must be in the UK currency format. The period must be used to indicate the decimal place. The comma must only be used to separate groups of thousands.
Currency	Yes	ISO4217	3 chars	ISO 4217 Examples: **GBP**, **EUR** and **USD**	The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.

Description	Yes	`<HTML>`	100 chars		Free text description of goods or services being purchased. This will be displayed on the Sage Pay Server payment page as the customer enters their card details.
CardHolder	Yes	Aa á /\ & - - "	50 chars		This should be the name displayed on the card.
CardNumber	Yes	0-9	20 chars		The full card number is required.
ExpiryDate	Yes	0-9	4 chars		The expiry date of the card in the format of **MMYY**
CV2	No	0-9	4 chars		The extra security 3 digits on the signature strip of the card, or the extra 4 digits on the front for American Express Cards If AVS/CV2 is ON for your account this field becomes compulsory.
CardType	Yes	Aa	15 chars	**VISA** **MC** **MCDEBIT** **DELTA** **MAESTRO** **UKE** **AMEX** **DC** **JCB**	**VISA** is Visa **MC** is MasterCard **MCDEBIT** is Debit MasterCard **DELTA** is Visa Debit **MAESTRO** is Domestic and International issued Maestro **UKE** is Visa Electron **AMEX** is American Express **DC** is Diners Club International and Discover **JCB** is Japan Credit Bureau The value should be in UPPERCASE.
BillingSurname	Yes	Aa á /\ & - - " , 0-9	20 chars		Customer billing details. All mandatory fields must contain a value, apart from the `BillingPostcode`. The `BillingPostcode` can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error.
BillingFirstnames	Yes	Aa á /\ & - - " , 0-9	20 chars		
BillingAddress1	Yes	Aa á /\ & - - " , 0-9 : + () CR/LF	100 chars		
BillingAddress2	No	Aa á /\ & - - " , 0-9 : + () CR/LF	100 chars		The `BillingState` becomes mandatory when the `BillingCountry` is set to **US**.

Field	Mandatory	Allowed characters	Length	Notes	
BillingCity	Yes	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 chars		
BillingPostCode	Yes	Aa - 0-9	10 chars		
BillingCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
BillingState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
BillingPhone	No	0-9 - Aa + ()	20 chars		
DeliverySurname	Yes	Aa á /\ & - . ' , 0-9	20 chars		Customer delivery details. All mandatory fields must contain a value, apart from the `DeliveryPostcode`. The `DeliveryPostcode` can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error. The `DeliveryState` becomes mandatory when the `DeliveryCountry` is set to **US**.
DeliveryFirstnames	Yes	Aa á /\ & - . ' , 0-9	20 chars		
DeliveryAddress1	Yes	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		
DeliveryAddress2	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		
DeliveryCity	Yes	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 chars		
DeliveryPostCode	Yes	Aa - 0-9	10 chars		
DeliveryCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
DeliveryState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
DeliveryPhone	No	0-9 - Aa + ()	20 chars		

CustomerEMail	No	RFC532N	255 chars	Examples: me@mail1.com:me@mail2.com	The customers email address.
					If you wish to use multiple email addresses, you should add them using the : (colon) character as a separator.
					The current version of the Direct integration method does not send confirmation emails to the customer. This field is provided for your records only.
Basket	No	<HTML>	7500 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the Basket field.	You can use this field to supply details of the customer's order. This information will be displayed to you in "My Sage Pay".
					If this field is supplied then the BasketXML field should not be supplied.
GiftAidPayment	No	0-9	Flag	**0** (default) **1**	Setting this field means the customer has ticked a box on your site to indicate they wish to donate the tax.
					0 = This transaction is not a Gift Aid charitable donation (default)
					1 = This payment is a Gift Aid charitable donation and the customer has AGREED to donate the tax.
					Only of use if your vendor account is Gift Aid enabled

ApplyAVSCV2	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the AVS/CV2 checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If AVS/CV2 enabled then check them. If rules apply, use rules (default) **1 =** Force AVS/CV2 checks even if not enabled for the account. If rules apply, use rules. **2 =** Force NO AVS/CV2 checks even if enabled on account. **3 =** Force AVS/CV2 checks even if not enabled for the account but DON'T apply any rules.
ClientIPAddress	No	0-9 .	15 chars		The IP address of the client connecting to your server making the payment. This should be a full IP address which you can obtain from your server scripts. We will attempt to Geolocate the IP address in your reports and fraud screening.

Apply3DSecure	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the 3D Secure checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If 3D-Secure checks are possible and rules allow, perform the checks and apply the authorisation rules. (default) **1 =** Force 3D-Secure checks for this transaction if possible and apply rules for authorisation. **2 =** Do not perform 3D-Secure checks for this transaction and always authorise. **3 =** Force 3D-Secure checks for this transaction if possible but ALWAYS obtain an auth code, irrespective of rule base.
AccountType	No	Aa	1 char	**E** (default) **M** **C**	This optional flag is used to tell the Sage Pay gateway which merchant account to use. If omitted, the system will use E, then M, then C by default. **E =** Use the e-commerce merchant account (default). **M =** Use the mail order/telephone order account (if present). **C =** Use the continuous authority merchant account (if present).

CreateToken	No	0-9	Flag	**0** (default) **1**	Use this flag to indicate you wish to have a token generated and stored in our database and returned to you for future use. **0 =** This will not create a token from the payment (default) **1 =** This will create a token from the payment if successful and return a `Token`.
BasketXML	No		20000 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the `BasketXML` field.	A more flexible version of the current basket field which can be used instead of the basket field. If this field is supplied then the Basket field should not be supplied.
CustomerXML	No		2000 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the `CustomerXML` field.	This can be used to supply information on the customer for purposes such as fraud screening.
SurchargeXML	No		800 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the `SurchargeXML` field.	Use this field to override current surcharge settings in "My Sage Pay" for the current transaction. Percentage and fixed amount surcharges can be set for different payment types.
VendorData	No	Aa 0-9	200 chars		Use this field to pass any data you wish to be displayed against the transaction in MySagePay.
ReferrerID	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 char		This can be used to send the unique reference for the Partner that referred the Vendor to Sage Pay.
Language	No	ISO639	2 chars	ISO 639-2 Examples: **EN**, **DE** and **FR**	The language the customer sees the payment pages in is determined by the code sent here. If this is not supplied then the language default of the customer's browser will be used. If the language is not supported then the language supported in the templates will be used. Currently supported languages in the Default templates are: French, German, Spanish, Portuguese, Dutch and English.

Website	No	Aa á /\ & - - ' , 0-9 : + () CR/LF	100 chars		Reference to the website this transaction came from. This field is useful if transactions can originate from more than one website. Supplying this information will enable reporting to be performed by website.
FIRecipientAcctNumber	No	Aa 0-9	10 chars		This should either be the first 6 and the last 4 characters of the primary recipient PAN (no spaces). Where the primary recipient account is not a card this will contain up to 10 characters of the account number (alphanumeric), unless the account number is less than 10 characters long in which case the account number will be present in its entirety. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientSurname	No	Aa	20 chars		This is the surname of the primary recipient. No special characters such as apostrophes or hyphens are permitted. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientPostcode	No	Aa 0-9			This is the postcode of the primary recipient. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientDoB	No	0-9			This is the date of birth of the primary recipient in the format YYYYMMDD This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)

Please refer to the Sage Pay Direct Integration and Protocol Guidelines 3.00 for the response to the above POST and subsequent POSTs required to complete the transaction process.

Appendix D: Standalone Token Registration – DIRECT integration

D1. You submit your token registration POST

This is performed via a HTTPS POST request, sent to the direct token registration URL service directtoken.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 -	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**TOKEN**	The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
Currency	Yes	ISO4217	3 chars	ISO 4217 Examples: **GBP**, **EUR** and **USD**	The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.
CardHolder	Yes	Aa á /\ & - ‚ '	50 chars		This should be the name displayed on the card.
CardNumber	Yes	0-9	20 chars		The full card number is required.
ExpiryDate	Yes	0-9	4 chars		The expiry date of the card in the format of **MMYY**
CV2	No	0-9	4 chars		The extra security 3 digits on the signature strip of the card, or the extra 4 digits on the front for American Express Cards If AVS/CV2 is ON for your account this field becomes compulsory.

CardType	Yes	Aa	15 chars	**VISA**	**VISA** is Visa
				MC	**MC** is MasterCard
				MCDEBIT	**MCDEBIT** is Debit MasterCard
				DELTA	**DELTA** is Visa Debit
				MAESTRO	**MAESTRO** is Domestic and International issued Maestro
				UKE	**UKE** is Visa Electron
				AMEX	**AMEX** is American Express
				DC	**DC** is Diners Club International and Discover
				JCB	**JCB** is Japan Credit Bureau
					The value should be in UPPERCASE.

D2. Sage Pay response to the token registration POST

This is the plain text response part of the POST originated by your servers in D1. Encoding will be as Name=Value pairs separated by carriage return and linefeeds (CRLF).

Response format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	`0-9` `.`	4 chars	**3.00**	Protocol version used by the system. Same as supplied in D1.
TxType	Yes	`Aa`	15 chars	**TOKEN**	Same as supplied in D1.
Token	No	`Aa` `0-9` `-` `{}`	38 chars		The token generated by Sage Pay.
Status	Yes	`Aa`	15 chars	**OK** **MALFORMED** **INVALID** **ERROR**	If the `Status` is not **OK**, the `StatusDetail` field will give more information about the problem. **OK =** Process executed without error. **MALFORMED =** Input message was missing fields or badly formatted – normally will only occur during development. **INVALID =** Transaction was not registered because although the POST format was valid, some information supplied was invalid. e.g. incorrect vendor name or currency. **ERROR =** A problem occurred at Sage Pay which prevented transaction registration. Please notify Sage Pay if a `Status` of **ERROR** is seen, together with your `Vendor`, `VendorTxCode` and the `StatusDetail`.
StatusDetail	No	`Aa` `0-9` `-` ` ` `()` `,` `:`	255 chars		Human-readable text providing extra detail for the `Status` message. Always check `StatusDetail` if the `Status` is not **OK**

Appendix E: Using a Token – SERVER integration

E1.   You submit your transaction registration POST

This is performed via a HTTPS POST request, sent to the initial Sage Pay Payment URL service vspserver-register.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 -	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**PAYMENT** **DEFERRED** **AUTHENTICATE**	See companion document "Server Integration and Protocol Guidelines 3.00" for more information on the different transaction types. The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
VendorTxCode	Yes	Aa 0-9 {} - _ -	40 chars		This should be your own reference code to the transaction. Your site should provide a completely unique VendorTxCode for each transaction.
Amount	Yes	0-9 - ,		0.01 to 100,000.00	Amount for the transaction containing minor digits formatted to 2 decimal places where appropriate. e.g. 5.10 or 3.29. Values such as 3.235 will be rejected. Minimum for no minor unit currencies like JPY is 1. Amounts must be in the UK currency format. The period must be used to indicate the decimal place. The comma must only be used to separate groups of thousands.
Currency	Yes	ISO4217	3 chars	ISO 4217 Examples: **GBP**, **EUR** and **USD**	The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.

Description	Yes	<HTML>	100 chars		Free text description of goods or services being purchased. This will be displayed on the Sage Pay Server payment page as the customer enters their card details.
NotificationURL	Yes	RFC1738	255 chars		This should be the fully qualified URL (including http:// or https:// header). It is the callback URL to which Notification POSTs are sent.
Token	Yes	Aa 0-9 -			The token provided during the token registration phase.
StoreToken	No	0-9	Flag	**0** (default) **1**	An optional flag to indicate if you wish to continue to store the Token in our token database for future use. **0 =** The Token will be deleted from our database regardless if it is authorised by the bank or not (default) **1 =** Continue to store the Token in our database for future use.
BillingSurname	Yes	Aa á /\ & - . ' , 0-9	20 chars		Customer billing details. All mandatory fields must contain a value, apart from the BillingPostcode. The BillingPostcode can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error. The BillingState becomes mandatory when the BillingCountry is set to **US**.
BillingFirstnames	Yes	Aa á /\ & - . ' , 0-9	20 chars		
BillingAddress1	Yes	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		
BillingAddress2	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		
BillingCity	Yes	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 chars		
BillingPostCode	Yes	Aa - 0-9	10 chars		
BillingCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
BillingState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	

Field	Mandatory	Characters	Length	Examples	Description
BillingPhone	No	0-9 - Aa + ( )	20 chars		
DeliverySurname	Yes	Aa á /\ & - ˋ ' , 0-9	20 chars		Customer delivery details. All mandatory fields must contain a value, apart from the DeliveryPostcode. The DeliveryPostcode can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error. The DeliveryState becomes mandatory when the DeliveryCountry is set to US.
DeliveryFirstnames	Yes	Aa á /\ & - ˋ ' , 0-9	20 chars		
DeliveryAddress1	Yes	Aa á /\ & - ˋ ' , 0-9 : + ( ) CR/LF	100 chars		
DeliveryAddress2	No	Aa á /\ & - ˋ ' , 0-9 : + ( ) CR/LF	100 chars		
DeliveryCity	Yes	Aa á /\ & - ˋ ' , 0-9 : + ( ) CR/LF	40 chars		
DeliveryPostCode	Yes	Aa - 0-9	10 chars		
DeliveryCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
DeliveryState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
DeliveryPhone	No	0-9 - Aa + ( )	20 chars		
CustomerEMail	No	RFC532N	255 chars	Examples: me@mail1.com:me@mail2.com	The customers email address. If you wish to use multiple email addresses, you should add them using the : (colon) character as a separator. The current version of the Server integration method does not send confirmation emails to the customer. This field is provided for your records only.
Basket	No	\<HTML\>	7500 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the Basket field.	You can use this field to supply details of the customer's order. This information will be displayed to you in "My Sage Pay". If this field is supplied then the BasketXML field should not be supplied.

AllowGiftAid	No	`0-9`	Flag	**0** (default) **1**	This flag allows the gift aid acceptance box to appear for this transaction on the payment page. This only appears if your vendor account is Gift Aid enabled. **0 =** No Gift Aid box displayed (default) **1 =** Display Gift Aid box on payment page.
ApplyAVSCV2	No	`0-9`	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the AVS/CV2 checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If AVS/CV2 enabled then check them. If rules apply, use rules (default) **1 =** Force AVS/CV2 checks even if not enabled for the account. If rules apply, use rules. **2 =** Force NO AVS/CV2 checks even if enabled on account. **3 =** Force AVS/CV2 checks even if not enabled for the account but DON'T apply any rules.

Apply3DSecure	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the 3D Secure checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If 3D-Secure checks are possible and rules allow, perform the checks and apply the authorisation rules. (default) **1 =** Force 3D-Secure checks for this transaction if possible and apply rules for authorisation. **2 =** Do not perform 3D-Secure checks for this transaction and always authorise. **3 =** Force 3D-Secure checks for this transaction if possible but ALWAYS obtain an auth code, irrespective of rule base.
Profile	No	Aa	10 chars	**NORMAL** (default) **LOW**	A profile of **LOW** returns the simplified payment pages which have only one step and minimal formatting. Designed to run in i-Frames. Omitting this field or sending **NORMAL** renders the normal card selection screen.
AccountType	No	Aa	1 char	**E** (default) **M** **C**	This optional flag is used to tell the Sage Pay gateway which merchant account to use. If omitted, the system will use E, then M, then C by default. **E =** Use the e-commerce merchant account (default). **M =** Use the mail order/telephone order account (if present). **C =** Use the continuous authority merchant account (if present).

BasketXML	No		20000 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the BasketXML field.	A more flexible version of the current basket field which can be used instead of the basket field. If this field is supplied then the Basket field should not be supplied.
CustomerXML	No		2000 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the CustomerXML field.	This can be used to supply information on the customer for purposes such as fraud screening.
SurchargeXML	No		800 chars	See companion document "Server Integration and Protocol Guidelines 3.00" for the format of the SurchargeXML field.	Use this field to override current surcharge settings in "My Sage Pay" for the current transaction. Percentage and fixed amount surcharges can be set for different payment types.
VendorData	No	Aa 0-9	200 chars		Use this field to pass any data you wish to be displayed against the transaction in "My Sage Pay".
ReferrerID	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	40 char		This can be used to send the unique reference for the Partner that referred the Vendor to Sage Pay.
Language	No	ISO639	2 chars	ISO 639-2 Examples: **EN**, **DE** and **FR**	The language the customer sees the payment pages in is determined by the code sent here. If this is not supplied then the language default of the shoppers browser will be used. If the language is not supported then the language supported in the templates will be used. Currently supported languages in the Default templates are: French, German, Spanish, Portuguese, Dutch and English.
Website	No	Aa á /\ & - . ' , 0-9 : + () CR/LF	100 chars		Reference to the website this transaction came from. This field is useful if transactions can originate from more than one website. Supplying this information will enable reporting to be performed by website.

`FIRecipientAcctNumber`	No	Aa 0-9	10 chars		This should either be the first 6 and the last 4 characters of the primary recipient PAN (no spaces). Where the primary recipient account is not a card this will contain up to 10 characters of the account number (alphanumeric), unless the account number is less than 10 characters long in which case the account number will be present in its entirety. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
`FIRecipientSurname`	No	Aa	20 chars		This is the surname of the primary recipient. No special characters such as apostrophes or hyphens are permitted. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
`FIRecipientPostcode`	No	Aa 0-9			This is the postcode of the primary recipient. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
`FIRecipientDoB`	No	0-9			This is the date of birth of the primary recipient in the format YYYYMMDD This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)

Please refer to the Sage Pay Server Integration and Protocol Guidelines 3.00 for the response to the above POST and subsequent POSTs required to complete the transaction process.

Appendix F: Using a Token – DIRECT integration

F1.   You submit your transaction registration POST

This is performed via a HTTPS POST request, sent to the initial Sage Pay Payment URL service vspdirect-register.vsp. The details should be URL encoded Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	`0-9` `.`	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	`Aa`	15 chars	**PAYMENT** **DEFERRED** **AUTHENTICATE**	See companion document "Direct Integration and Protocol Guidelines 3.00" for more information on the different transaction types. The value should be in UPPERCASE.
Vendor	Yes	`Aa` `0-9`	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
VendorTxCode	Yes	`Aa` `0-9` `{}` `.` `_` `-`	40 chars		This should be your own reference code to the transaction. Your site should provide a completely unique `VendorTxCode` for each transaction.
Amount	Yes	`0-9` `.` `,`		0.01 to 100,000.00	Amount for the transaction containing minor digits formatted to 2 decimal places where appropriate. e.g. 5.10 or 3.29. Values such as 3.235 will be rejected. Minimum for no minor unit currencies like JPY is 1. Amounts must be in the UK currency format. The period must be used to indicate the decimal place. The comma must only be used to separate groups of thousands.
Currency	Yes	`ISO4217`	3 chars	ISO 4217 Examples: **GBP**, **EUR** and **USD**	The currency the transaction is performed in. This must be supported by one of your Sage Pay merchant accounts or the transaction will be rejected.

Field	Required	Chars	Size	Values	Description
Description	Yes	`<HTML>`	100 chars		Free text description of goods or services being purchased. This will be displayed on the Sage Pay Server payment page as the customer enters their card details.
Token	Yes	`Aa 0-9 -`			The token provided during the token registration phase.
StoreToken	No	`0-9`	Flag	**0** (default) **1**	An optional flag to indicate if you wish to continue to store the `Token` in our token database for future use. **0 =** The `Token` will be deleted from our database if authorised by the bank (default) **1 =** Continue to store the `Token` in our database for future use.
CV2	No	`0-9`	4 chars		The extra security 3 digits on the signature strip of the card, or the extra 4 digits on the front for American Express Cards If AVS/CV2 is ON for your account this field becomes compulsory.
BillingSurname	Yes	`Aa á /\ & - . '`	20 chars		Customer billing details. All mandatory fields must contain a value, apart from the postcode. The postcode can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error.
BillingFirstnames	Yes	`Aa á /\ & - . '`	20 chars		
BillingAddress1	Yes	`Aa á /\ & - . ' , 0-9 : + () CR/LF`	100 chars		
BillingAddress2	No	`Aa á /\ & - . ' , 0-9 : + () CR/LF`	100 chars		
BillingCity	Yes	`Aa á /\ & - . ' , 0-9 : + () CR/LF`	40 chars		
BillingPostCode	Yes	`Aa - 0-9`	10 chars		
BillingCountry	Yes	`ISO3166`	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	

Field	Mandatory	Type	Length	Values	Description
BillingState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
BillingPhone	No	0-9 − Aa + ( )	20 chars		
DeliverySurname	Yes	Aa á /\ & − . '	20 chars		Customer delivery details. All mandatory fields must contain a value, apart from the postcode. The postcode can be blank for countries that do not have postcodes (e.g. Ireland) but is required in all countries that do have them. Providing a blank field when information is required will cause an error.
DeliveryFirstnames	Yes	Aa á /\ & − . '	20 chars		
DeliveryAddress1	Yes	Aa á /\ & − . ' , 0-9 : + ( ) CR/LF	100 chars		
DeliveryAddress2	No	Aa á /\ & − . ' , 0-9 : + ( ) CR/LF	100 chars		
DeliveryCity	Yes	Aa á /\ & − . ' , 0-9 : + ( ) CR/LF	40 chars		
DeliveryPostCode	Yes	Aa − 0-9	10 chars		
DeliveryCountry	Yes	ISO3166	2 chars	ISO 3166 Examples: **GB**, **IE** and **DE**	
DeliveryState	No	US	2 chars	Examples: **AL**, **MS** and **NY**	
DeliveryPhone	No	0-9 − Aa + ( )	20 chars		
CustomerEMail	No	RFC532N	255 chars	Examples: me@mail1.com:me@mail2.com	The customers email address. If you wish to use multiple email addresses, you should add them using the : (colon) character as a separator. The current version of the Direct integration method does not send confirmation emails to the customer. This field is provided for your records only.
Basket	No	<HTML>	7500 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the `Basket` field.	You can use this field to supply details of the customer's order. This information will be displayed to you in "My Sage Pay". If this field is supplied then the `BasketXML` field should not be supplied.

GiftAidPayment	No	0-9	Flag	**0** (default) **1**	Setting this field means the customer has ticked a box on your site to indicate they wish to donate the tax. **0** = This transaction is not a Gift Aid charitable donation (default) **1** = This payment is a Gift Aid charitable donation and the customer has AGREED to donate the tax. Only of use if your vendor account is Gift Aid enabled
ApplyAVSCV2	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the AVS/CV2 checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If AVS/CV2 enabled then check them. If rules apply, use rules (default) **1 =** Force AVS/CV2 checks even if not enabled for the account. If rules apply, use rules. **2 =** Force NO AVS/CV2 checks even if enabled on account. **3 =** Force AVS/CV2 checks even if not enabled for the account but DON'T apply any rules.
ClientIPAddress	No	0-9 .	15 chars		The IP address of the client connecting to your server making the payment. This should be a full IP address which you can obtain from your server scripts. We will attempt to Geolocate the IP address in your reports and fraud screening.

Apply3DSecure	No	0-9	Flag	**0** (default) **1** **2** **3**	Using this flag you can fine tune the 3D Secure checks and rule set you've defined at a transaction level. This is useful in circumstances where direct and trusted customer contact has been established and you wish to override the default security checks. **0 =** If 3D-Secure checks are possible and rules allow, perform the checks and apply the authorisation rules. (default) **1 =** Force 3D-Secure checks for this transaction if possible and apply rules for authorisation. **2 =** Do not perform 3D-Secure checks for this transaction and always authorise. **3 =** Force 3D-Secure checks for this transaction if possible but ALWAYS obtain an auth code, irrespective of rule base.
AccountType	No	Aa	1 char	**E** (default) **M** **C**	This optional flag is used to tell the Sage Pay gateway which merchant account to use. If omitted, the system will use E, then M, then C by default. **E =** Use the e-commerce merchant account (default). **M =** Use the mail order/telephone order account (if present). **C =** Use the continuous authority merchant account (if present).
BasketXML	No		20000 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the BasketXML field.	A more flexible version of the current basket field which can be used instead of the basket field. If this field is supplied then the Basket field should not be supplied.

CustomerXML	No		2000 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the CustomerXML field.	This can be used to supply information on the customer for purposes such as fraud screening.
SurchargeXML	No		800 chars	See companion document "Direct Integration and Protocol Guidelines 3.00" for the format of the SurchargeXML field.	Use this field to override current surcharge settings in MySagePay for the current transaction. Percentage and fixed amount surcharges can be set for different payment types.
VendorData	No	Aa 0-9	200 chars		Use this field to pass any data you wish to be displayed against the transaction in "My Sage Pay".
ReferrerID	No	Aa á /\ & - - ' , 0-9 : + () CR/LF	40 char		This can be used to send the unique reference for the Partner that referred the Vendor to Sage Pay.
Language	No	ISO639	2 chars	ISO 639-2 Examples: **EN**, **DE** and **FR**	The language the customer sees the payment pages in is determined by the code sent here. If this is not supplied then the language default of the shoppers browser will be used. If the language is not supported then the language supported in the templates will be used. Currently supported languages in the Default templates are: French, German, Spanish, Portuguese, Dutch and English.
Website	No	Aa á /\ & - - ' , 0-9 : + () CR/LF	100 chars		Reference to the website this transaction came from. This field is useful if transactions can originate from more than one website. Supplying this information will enable reporting to be performed by website.

FIRecipientAcctNumber	No	Aa 0-9	10 chars		This should either be the first 6 and the last 4 characters of the primary recipient PAN (no spaces). Where the primary recipient account is not a card this will contain up to 10 characters of the account number (alphanumeric), unless the account number is less than 10 characters long in which case the account number will be present in its entirety. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientSurname	No	Aa	20 chars		This is the surname of the primary recipient. No special characters such as apostrophes or hyphens are permitted. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientPostcode	No	Aa 0-9			This is the postcode of the primary recipient. This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)
FIRecipientDoB	No	0-9			This is the date of birth of the primary recipient in the format YYYYMMDD This field is only required for UK merchants who have a merchant category code of 6012 (Financial Institutions)

Please refer to the Sage Pay Direct Integration and Protocol Guidelines 3.00 for the response to the above POST and subsequent POSTs required to complete the transaction process.

Appendix G: Removing a Token

## G1.	You submit your token removal POST

This is performed via HTTPS POSTs sent to the remove token URL service removetoken.vsp. The details must be URL encoded, with Name=Value fields separated by '&' characters.

Request format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 -	4 chars	**3.00**	This is the version of the protocol you are integrating with. Default or incorrect value is taken to be **3.00**.
TxType	Yes	Aa	15 chars	**REMOVETOKEN**	The value should be in UPPERCASE.
Vendor	Yes	Aa 0-9	15 chars		Used to authenticate your site. This should contain the Sage Pay Vendor Name supplied by Sage Pay when your account was created.
Token	Yes	Aa 0-9 -			The token provided during the token registration phase.

G2. Server response to the token removal POST

This is the plain text response part of the POST originared by your servers in G1. Encoding will be as Name=Value pairs separated by carriage return and linefeeds (CRLF).

Response format

Name	Mandatory	Format	Max Length	Allowed Values	Description
VPSProtocol	Yes	0-9 .	4 chars	**3.00**	Protocol version used by the system. Same as supplied in G1.
Status	Yes	Aa	15 chars	**OK** **MALFORMED** **INVALID** **ERROR** .	If the Status is not **OK**, the StatusDetail field will give more information about the problem. **OK =** Process executed without error. **MALFORMED =** Input message was missing fields or badly formatted – normally will only occur during development. **INVALID =** Transaction was not registered because although the POST format was valid, some information supplied was invalid. e.g. incorrect vendor name or currency. **ERROR =** A problem occurred at Sage Pay which prevented transaction registration. Please notify Sage Pay if a Status of **ERROR** is seen, together with your Vendor, VendorTxCode and the StatusDetail.
StatusDetail	Yes	Aa 0-9 . () , :	255 chars		Human-readable text providing extra detail for the Status message. Always check StatusDetail if the Status is not **OK**

7.0 URLs

Part of a Transaction Token Registration

Integration	Environment	URL
SERVER	TEST	https://test.sagepay.com/gateway/service/vspserver-register.vsp
SERVER	LIVE	https://live.sagepay.com/gateway/service/vspserver-register.vsp
DIRECT	TEST	https://test.sagepay.com/gateway/service/vspdirect-register.vsp
DIRECT	LIVE	https://live.sagepay.com/gateway/service/vspdirect-register.vsp

Standalone Token Registration

Integration	Environment	URL
SERVER	TEST	https://test.sagepay.com/gateway/service/token.vsp
SERVER	LIVE	https://live.sagepay.com/gateway/service/token.vsp
DIRECT	TEST	https://test.sagepay.com/gateway/service/directtoken.vsp
DIRECT	LIVE	https://live.sagepay.com/gateway/service/directtoken.vsp

Using a Token

Integration	Environment	URL
SERVER	TEST	https://test.sagepay.com/gateway/service/vspserver-register.vsp
SERVER	LIVE	https://live.sagepay.com/gateway/service/vspserver-register.vsp
DIRECT	TEST	https://test.sagepay.com/gateway/service/vspdirect-register.vsp
DIRECT	LIVE	https://live.sagepay.com/gateway/service/vspdirect-register.vsp

Removing a Token

Integration	Environment	URL
SERVER & DIRECT	TEST	https://test.sagepay.com/gateway/service/removetoken.vsp
SERVER & DIRECT	LIVE	https://live.sagepay.com/gateway/service/ removetoken.vsp