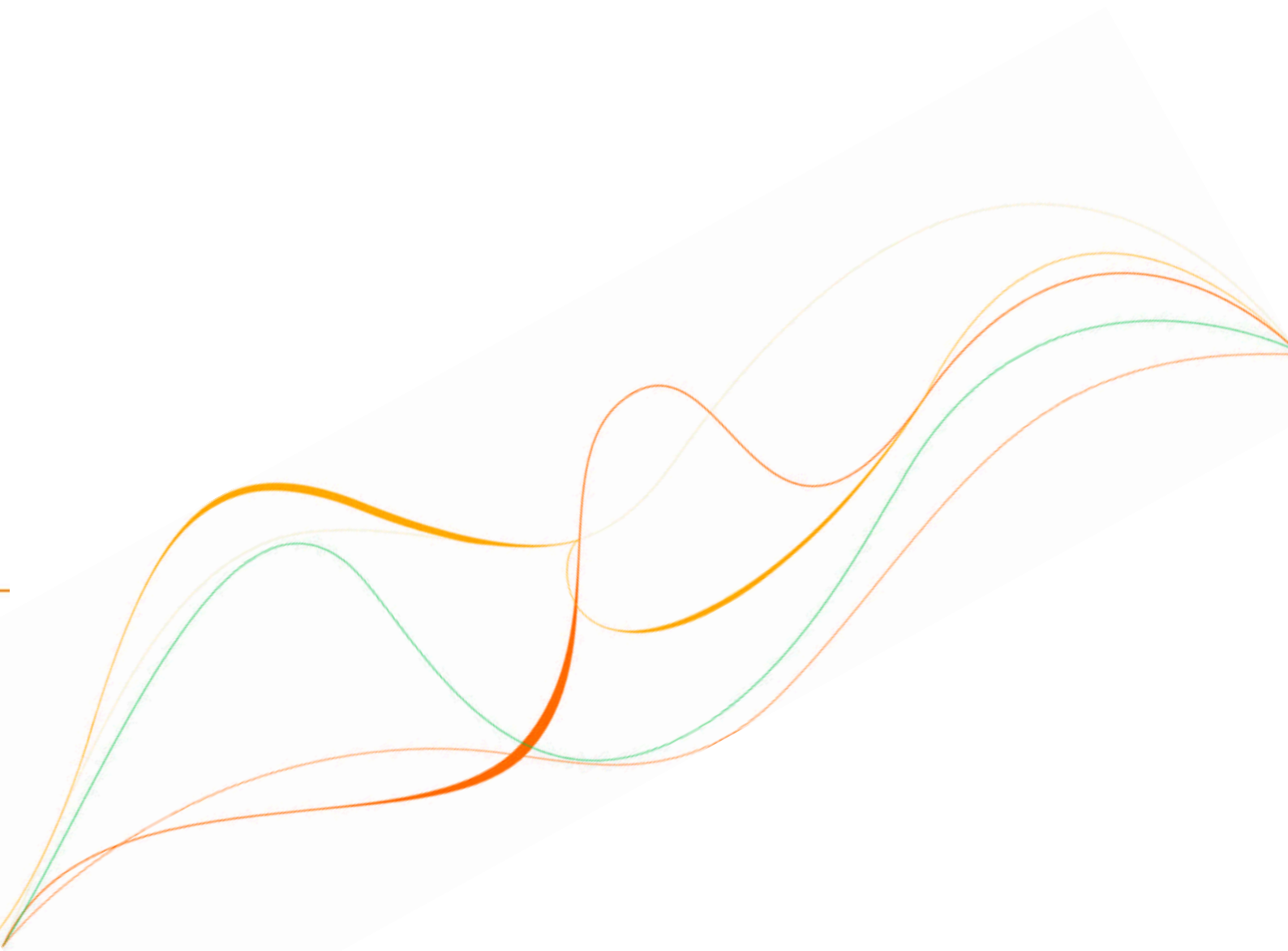




Reporting & Admin API Integration Guideline

Protocol version 1.02



Content

1	Introduction.....	1
1.1	How to use this integration guide.....	1
1.2	API Request and Response Formats.....	1
1.3	When no XML field is supplied or <command> element is "version".....	2
2	The different <command> options.....	3
2.1	Admin Command - version.....	5
2.2	Admin Command - createUser.....	6
2.3	Admin Command - deleteUser.....	8
2.4	Admin Command - lockUser.....	9
2.5	Admin Command - logAccountActivity.....	10
2.6	Admin Command - loginVendorUser.....	11
2.7	Admin Command - logoutUser.....	12
2.8	Admin Command - unlockUser.....	13
2.9	Admin Command - setUserPassword.....	14
2.10	Admin Command - setUserRights.....	15
2.11	Admin Command - set3DSecureStatus.....	17
2.12	Admin Command - setAVSCV2Status.....	18
2.13	Admin Command - set3DSecureRules.....	19
2.14	Admin Command - setAVSCV2Rules.....	20
2.15	Admin Command - setExpiryDate.....	21
2.16	Admin Command - setVendorDetails.....	23
2.17	Admin Command - addBlockedCardRanges.....	24
2.18	Admin Command - addBlockedCountries.....	25
2.19	Admin Command - addBlockedIPs.....	26
2.20	Admin Command - addBlockedIssuingCountries.....	27
2.21	Admin Command - addValidIPs.....	28
2.22	Admin Command - deleteAVSCV2Rules.....	29
2.23	Admin Command - delete3DSecureRules.....	30
2.24	Admin Command - deleteBlockedCardRanges.....	31
2.25	Admin Command - deleteBlockedCountries.....	32
2.26	Admin Command - deleteBlockedIPs.....	33
2.27	Admin Command - deleteBlockedIssuingCountries.....	34
2.28	Admin Command - deleteValidIPs.....	35
2.29	Admin Command - setPaymentPageTemplates.....	36
2.30	Admin Command - setPaypalAccount.....	37
2.31	Admin Command - deletePaypalAccount.....	38
2.32	Reporting Command - doesUserExist.....	39
2.33	Reporting Command - getUserRights.....	40
2.34	Reporting Command - getAccountLogs.....	42
2.35	Reporting Command - getVendorUserList.....	43
2.36	Reporting Command - getVendorDetails.....	45
2.37	Reporting Command - getVendorPaymentSystems.....	47
2.38	Reporting Command - getValidIPs.....	48
2.39	Reporting Command - getURLs.....	49
2.40	Reporting Command - get3DSecureStatus.....	51
2.41	Reporting Command - get3DSecureRules.....	52
2.42	Reporting Command - getAVSCV2Status.....	53
2.43	Reporting Command - getAvsCv2Rules.....	54
2.44	Reporting Command - getBatchDetail.....	56
2.45	Reporting Command - getBatchList.....	59
2.46	Reporting Command - getCardType.....	61
2.47	Reporting Command - getCardDetails.....	62
2.48	Reporting Command - getCurrencies.....	64
2.49	Reporting Command - getMerchantAccounts.....	65
2.50	Reporting Command - getTransactionCardDetails.....	67
2.51	Reporting Command - getTransactionDetail.....	69
2.52	Reporting Command - getTransactionIPDetails.....	75
2.53	Reporting Command - getTransactionList.....	77
2.54	Reporting Command - getTransactionSummary.....	82
2.55	Reporting Command - getTransactionTypes.....	85
2.56	Reporting Command - getRelatedTransactions.....	86
2.57	Reporting Command - getGiftAidReport.....	89
2.58	Reporting Command - getAllAuthProcessors.....	91
2.59	Reporting Command - getAllCountries.....	92
2.60	Reporting Command - getBlockedCardRanges.....	93
2.61	Reporting Command - getBlockedCountries.....	94

2.62	Reporting Command - getBlockedIPs	95
2.63	Reporting Command - getBlockedIssuingCountries	96
2.64	Reporting Command - getTokenCount	97
2.65	Reporting Command - getTokenDetails	98
2.66	Reporting Command - getT3MDetail	100
3	Using Special Characters	102
4	Full <errorcode> list	102

1 Introduction

This document details the full set of requests and responses available from the Reporting & Admin API system. This API allows your back-office systems to interact with the Sage Pay payment systems in the same manner as a person with a web browser might use our My Sage Pay administration area.

The API uses the same user accounts and privilege system as My Sage Pay, and will allow account administration and reporting functions, but with a finer degree of granularity than the My Sage Pay Screens, with all messages in real-time in a consistent XML format.

All requests are sent to one gateway at the following address:

Test Server: <https://test.sagepay.com/access/access.htm>

Live Server: <https://live.sagepay.com/access/access.htm>

1.1 How to use this integration guide

All the various commands as listed within this guide and outlined in the same structure. You will find a brief overview of what the command is for along with both the request and response protocol when using the command. Along with the protocol, you will find some example codes of the request and response to help you.

Please beware of the following symbols throughout this documentation:



The user name provided must have administrative privileges to run this command.



This command doesn't require the user to have administrator rights.

1.2 API Request and Response Formats

The API requests are HTTPS POSTs to the URLs above, containing a single FORM field called **XML**. **It is important to note that these are not requests in XML format, but regular HTTPS POSTs with the content type "application/x-www-form-urlencoded" with the XML requests detailed below in a FORM field called XML**

The XML field will contain the XML message, which always takes the following format:

```
<vspaccess>
  <command>whatever</command>
  <vendor>Vendor Name</vendor>
  <user>User name</user>
  <other command specific parameters in here..../>
  <signature>MD5 Hash Signature</signature>
</vspaccess>
```

The **<signature>** value is created by taking the rest of the contents of the **<vspaccess>** node, appending an element called **<password>** which contains the user account's password, and MD5 hashing the message. So in the above example:

```
MD5 Hash Signature =  
MD5Hash(  
    "<command>whatever</command>  
    <vendor>VSP Vendor Name</vendor>  
    <user>User name</user>  
    <other command specific params in here..../>  
    <password>user password</password>")
```

The response format will always be a XML v1.0 ISO-8859-1 encoded document with a root node called **<vspaccess>**. i.e.:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<vspaccess>  
    <errorcode>nnnn</errorcode>  
    <error>Text error message if errorcode is not 0000</error>  
    <other command specific results here..../>  
    <timestamp>DD/MM/YY HH:MM:SS GMT</timestamp>  
</vspaccess>
```

The **<errorcode>** field is always present, and for successful requests will always be **0000**. All other values indicate an error, and the **<error>** element will provide a human readable interpretation of the error code. A list of error codes is given in Appendix A below.

The **<timestamp>** will also always be present and contains the time that the response was sent back to the requesting process.

1.3 When no XML field is supplied or **<command> element is "version"**

If no XML field is supplied in the POST, or if the **<command>** element is set to "version" (i.e. **<command>version</command>**) the system will simply return the version number of the active Access system, and the timestamp. For example:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<vspaccess>  
    <version>1.02</version>  
    <timestamp>25/03/09 15:16:50 GMT</timestamp>  
</vspaccess>
```

2 The different <command> options

This section lists the commands that can be passed in the <command> element of the request. This list will grow with subsequent releases, so please ensure you are working from the most recent release of this document.

Each section lists the command and the required and optional parameters that form the complete Request in the table below it. This table should be mapped to valid API XML request by appending the <signature> element and <vspaccess> node as described in section 1.1 above. The Response from the API, again with optional components, is listed immediately after.

For example, the **getURLs** command has the following request format:

Node/Element	Type	Description
<command>	String getURLs	The service being requested.
<vendor>	String (16 characters)	The VSP Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name under the above referenced Vendor account.
<system>	String (10 characters)	The system for which the URLs are being requested. Must be either VSPServer , VSPDirect or VSPForm
<txtypes> (optional)	Node	Container for the list of transactions types for which the URLs are being requested. If not provided, all service URLs are returned.
L<txtype> (optional)	String (20 characters)	List of transaction types for which the URLs are being requested.

This would translate to an XML request like this:

```
<vspaccess>
  <command>getURLs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <system>VSPDirect</system>
  <txtypes>
    <txtype>payment</txtype>
    <txtype>refund</txtype>
  </txtypes>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

This would request the URLs for the payment and refund services for the Direct payment system.

The Response to that request is summarised in a table like this:

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<urls>	Node	Container for the URLs being returns
L<url>	Node	Container for a single URL with the service it matches
L<txtype>	String (20 characters)	The name of the transaction type for which this URL maps
L<serviceURL>	String - FQDN (255 characters)	Fully Qualified Domain Name URL to the service named above.

This would map to the following response in valid XML:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<vspaccess>
  <errorcode>0000</errorcode>
  <urls>
    <url>
      <txtype>payment</txtype>
      <serviceURL>https://test.sagepay.com/payment</serviceURL>
    </url>
    <url>
      <txtype>refund</txtype>
      <serviceURL>https://test.sagepay.com/refund</serviceURL>
    </url>
  </urls>
  <timestamp>25/03/09 15:16:50 GMT</timestamp>
</vspaccess>
```

2.1 Admin Command - version

This command returns the version number of the Access system to which the message is sent. This command is executed by default if no command is specified.

Unlike all other commands, a vendor, user and signature are not required to obtain a response from the version request.

Request

Node/Element	Type	Description
<command>	String version	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<version>	Alphanumeric (10 characters)	The current version number of the Access system being accessed.

Sample Input XML:

```
<vspaccess>
  <command>version</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
</vspaccess>
```

Sample output XML:

```
<vspaccess>
  <errorcode>0000</errorcode>
  <version>1.02</version>
</vspaccess>
```


2.2 Admin Command - createUser

The createUser command allows the creation of a user account accessible to both My Sage Pay and this API. It mimics the user account creation process in the Administration section of the My Sage Pay screens.



The user name provided must have administrative privileges to run this command.

See also: deleteUser, setUserRights, lockUser, unlockUser, logoutUser, setUserPassword

Request

Node/Element	Type	Description
<command>	String createUser	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command. Must Have ADMIN rights on the account.
<newuser>	Node	Container for the new user account parameters.
L<username>	Alphanumeric (20 characters)	The new user's user name. Can only contain letters and numbers.
L<password>	String (32 characters)	The new user's password. CASE SENSITIVE and can contain ASCII characters
L<accessrights>	Node	Container for the access rights
L<access>	String (specific values)	Access values can be: Transactions – Access to the transaction area Reports – Access to the reporting area Terminal – Can process Terminal transactions Admin – USER HAS ADMIN RIGHTS! CanViewAll – Can view other user's transactions as well as their own. CanRefund, CanRelease, CanAbort, CanRepeat, CanVoid and CanManual – User can perform these transaction types if passed
L<homepage> (optional)	String (20 characters)	The initial page the user is presented with when they log in. The following values can be used: transactions.msp – the report generation page settings.msp – the account administration page terminal.msp – the virtual terminal page search.msp – search transactions

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>createUser</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <newuser>
    <username>steve</username>
    <password>mypassword</password>
    <accessrights>
      <access>TRANSACTIONS</access>
      <access>REPORTS</access>
    </accessrights>
    <homepage>settings.msp</homepage>
  </newuser>
  <signature>F0B8EBE0C25BF6D1496F7E082D525D4A</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.3 Admin Command - deleteUser

The deleteUser command allows the removal of a user from both My Sage Pay and this API. It mimics the user deletion process in the Administration section of the My Sage Pay screens.



The user name provided must have administrative privileges to run this command.

See also: createUser, setUserRights, lockUser, unlockUser, logoutUser, setPassword

Request

Node/Element	Type	Description
<command>	String deleteUser	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The username of the user to be deleted.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteUser</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <username>johnsmith</username>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.4 Admin Command - lockUser

This command will lock out a specified user account until an administrator unlocks it. It will also log out the specific user if they are active at the time this command is issued.



The user name provided must have administrative privileges to run this command.

See also: createUser, deleteUser, setUserRights, unlockUser, logoutUser, setPassword

Request

Node/Element	Type	Description
<command>	String lockUser	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user account to lock out.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>lockUser</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <username>johnsmith</username>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.5 Admin Command - logAccountActivity

This command sets an account log for changes to the VSPReporting database.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String logAccountActivity	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	String (20 characters)	The name of the user completing the account activity
<logentry>	String (276 characters)	The account activity to be logged

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>getAccountLogs</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <username>dave</username>
  <logentry>ip address added</logentry>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
</vspaccess>
```

2.6 Admin Command - loginVendorUser

Logs in the specified user into the vendor account provided, allowing them access to any area of the system they have been granted the right to use.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String loginVendorUser	The service being requested.
<vendor>	String (16 characters)	The vendor account on which the user will be logged in.
<user>	String (20 characters)	Must be the admin user under the vendor account
<username>	String (20 characters)	The user you wish to login.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>loginVendorUser</command>
  <vendor>egoo</vendor>
  <user>egoo</user>
  <username>John</username>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <vendorid>8117BOC2-899A-F3B5-2A3B-67A716DE3BBE</vendorid>
  <vendorprovidedname/>
  <sessionid>USR975309759262545</sessionid>
  <status>0</status>
  <homepage>search.msp</homepage>
  <timestamp>20/10/2010 11:39:13</timestamp>
</vspaccess>
</response>
```

2.7 Admin Command - logoutUser

Use the logoutUser command to terminate the current session for the specified user. MySagePay will automatically time-out user accounts after 15 minutes of inactivity, but this command can be used to force a logout if necessary.



The user name provided must have administrative privileges to run this command.

See also: createUser, deleteUser, setUserRights, lockUser, unlockUser, setUserPassword

Request

Node/Element	Type	Description
<command>	String logoutUser	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user account to log out.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML input:

```
<vspaccess>
  <command>logoutUser</command>
  <vendor>sagepay</vendor>
  <user>charlie</user>
  <username>AnotherUserName</username>
</vspaccess>
```

Response XML Output:

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.8 Admin Command - unlockUser

Use this command to unlock user accounts that have been locked out either by three incorrect login attempts, or by the account being specifically locked by an administrator.



The user name provided must have administrative privileges to run this command.

See also: createUser, deleteUser, setUserRights, lockUser, logoutUser, setUserPassword

Request

Node/Element	Type	Description
<command>	String unlockUser	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user account to unlock.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>unlockUser</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <username>johnsmith</username>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```


2.9 Admin Command - setUserPassword

This command can be used to overwrite the existing user's password with the new password specified. This is useful if the user has forgotten their password.



The user name provided must have administrative privileges to run this command.

See also: createUser, deleteUser, setUserRights, lockUser, unlockUser, logoutUser

Request

Node/Element	Type	Description
<command>	String setUserPassword	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user account requiring the new password. The username cannot be the same as the vendor (i.e. it cannot be the admin account)
<password>	String (32 characters)	The new user's password. CASE SENSITIVE and can contain ASCII characters

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>setUserPassword</command>
  <vendor>onlinecheese</vendor>
  <user>sagepay</user>
  <username>sagepay</username>
  <password>password123</password>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.10 Admin Command - setUserRights

The setUserRights command allows you to overwrite the privileges assigned to the given MySagePay and Access user. This is equivalent to editing the user in MySagePay. NB: This command will REMOVE ALL EXISTING access rights and replace them with those specified. It does NOT add to the existing rights.



The user name provided must have administrative privileges to run this command.

See also: createUser, deleteUser, lockUser, unlockUser, logoutUser, setUserPassword

Request

Node/Element	Type	Description
<command>	String setUserRights	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user account requiring the access right changes. The username cannot be the same as the vendor (i.e. it cannot be the admin account)
<accessrights>	Node	Container for the access rights
^L <access>	String (specific values)	Access values can be: Transactions – Access to the transaction area Reports – Access to the reporting area Terminal – Can process Terminal transactions Admin – USER HAS ADMIN RIGHTS! CanViewAll – Can view other user's transactions as well as their own. CanRefund, CanRelease, CanAbort, CanRepeat, CanVoid and CanManual – User can perform these transaction types if passed
<homepage>	String (20 characters)	The initial page the user is presented with when they log in. The following values can be used: transactionrefine.msp – the search page transactions.msp – the report generation page settings.msp – the account administration page terminal.msp – the virtual terminal page search.msp – search transactions

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Example request on overleaf..

Sample XML Input

```
<vspaccess>
  <command>setUserRights</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <username>bobsmith</username>
  <accessrights>
    <access>Terminal</access>
  </accessrights>
  <homepage>terminal.msp</homepage>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</ errorcode >
</vspaccess>
```

2.11 Admin Command - set3DSecureStatus

This command allows you to switch 3D Secure checks on or off for the chosen vendor. You cannot remove or add rules with this command, so switching on 3D Secure will also turn on existing 3D Secure rules (if they exist on this account)



The user name provided must have administrative privileges to run this command.

See also: getAVSCV2Status, setAVSCV2Status, get3DSecureStatus

Request

Node/Element	Type	Description
<command>	String set3DSecureStatus	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<status>	String (3 characters)	Should contain either ON or OFF to indicate the preferred status of the 3D-Secure checks.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Output

```
<vspaccess>
  <command>set3DSecureStatus</command>
  <vendor>sagepay</vendor>
  <user>charlie</user>
  <status>ON</status>
</vspaccess>
```

Sample XML Output

```
<errorcode>0000</errorcode>
```

2.12 Admin Command - setAVSCV2Status

This command allows you to switch AVS/CV2 checks on or off for the chosen vendor. You cannot remove or add rules with this command, so switching on AVS/CV2 will also turn on existing AVS/CV2 rules (if they exist on this account)



The user name provided must have administrative privileges to run this command.

See also: get3DSecureStatus, getAVSCV2Status, set3DSecureStatus

Request

Node/Element	Type	Description
<command>	String setAVSCV2Status	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<status>	String (3 characters)	Should contain either ON or OFF to indicate the preferred status of the AVS/CV2 checks.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>etAVSCV2Status</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <status>ON</status>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.13 Admin Command - set3DSecureRules

This command sets up rule set for the 3D Secure authentication system.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String set3DSecureRules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startvalue>	Numeric (monitary value)	The starting value you wish the particular rule to apply from (must not overlap existing rules)
<endvalue>	Numeric (monitary value)	The finishing value you wish the particular rule to apply to (must not overlap existing rules)
<perform3dauth>	String (3 characters)	YES or NO
<allownon3dcards>	String (3 characters)	YES or NO
<allownon3dissuers>	String (3 characters)	YES or NO
<allow3dfailures>	String (3 characters)	YES or NO
<allowmpifailures>	String (3 characters)	YES or NO

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>set3dSecureRules</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <startvalue>0.00</startvalue>
  <endvalue>1000.00</endvalue>
  <perform3dauth>YES</perform3dauth>
  <allownon3dcards>YES</allownon3dcards>
  <allownon3dissuers>YES</allownon3dissuers>
  <allow3dfailures>YES</allow3dfailures>
  <allowmpifailures>YES</allowmpifailures>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
</vspaccess>
```

2.14 Admin Command - setAVSCV2Rules

This command sets up a rule set for the AvsCv2 authentication system.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String setAVSCV2Rules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startvalue>	Numeric (monitary value)	The starting value you wish the particular rule to apply from (must not overlap existing rules)
<endvalue>	Numeric (monitary value)	The finishing value you wish the particular rule to apply to (must not overlap existing rules)
<allownodatamatches>	String (3 characters)	YES or NO
<allowaddressmatchonly>	String (3 characters)	YES or NO
<allowsecuritycodematchonly>	String (3 characters)	YES or NO
<allowdatanotchecked>	String (3 characters)	YES or NO

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>setAVSCV2Rules</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <startvalue>0.00</startvalue>
  <endvalue>1000.00</endvalue>
  <allownodatamatches>YES</allownodatamatches>
  <allowaddressmatchonly>YES</allowaddressmatchonly>
  <allowsecuritycodematchonly>YES</allowsecuritycodematchonly>
  <allowdatanotchecked>YES</allowdatanotchecked>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.15 Admin Command - setExpiryDate

This command allows the expiry date of the card used in the specified transaction, to be changed. This is useful for continuous authority payments where cards have been re-issued but the card number has not been changed. This modified transaction can then be REPEATED against without authorisation failures.

The transactions can be specified by either VendorTxCode or VPSTxID. The new ExpiryDate must be in MMY (two digit month and two digit year) format.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String setExpiryDate	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<vendortxcode> (optional) OR	String (40 characters)	The VendorTxCode of the transaction
<vpstxid> (optional)	String (40 characters)	The VPSTxID (transactionid) of the transaction
<startdate> (optional)	String (4 characters)	MMYY format e.g. 0307 for March 2007
<expirydate>	String (4 characters)	MMYY format e.g. 0309 for March 2009

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Example request on overleaf..

Example input XML 1:

```
<vspaccess>
  <command>setExpiryDate</command>
  <vendor>egoo</vendor>
  <user>Charlie</user>
  <vendortxcode>12345-CUSTOMER123</vendortxcode>
  <expirydate>0111</expirydate>
</vspaccess>
```

Example input XML 2:

```
<vspaccess>
  <command>setExpiryDate</command>
  <vendor>egoo</vendor>
  <user>Charlie</user>
  <vpstxid>AF231321-3821903D-3712379123712</vpstxid>
  <expirydate>0111</expirydate>
</vspaccess>
```

Example input XML 3:

```
<vspaccess>
  <command>setExpiryDate</command>
  <vendor>egoo</vendor>
  <user>Charlie</user>
  <vendortxcode>12345-CUSTOMER123</vendortxcode>
  <startdate>0108</startdate>
  <expirydate>0111</expirydate>
</vspaccess>
```

Example Output:

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.16 Admin Command - setVendorDetails

Using this command will set the vendor's account details.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String	The service being requested.
<vendor>	String (16 characters)	Must be protx
<user>	String (20 characters)	Must be an admin user under the Sage Pay account
<vendordetails>	Node	A container for the vendor details elements.
└<vendorprovidedname>	String (50 characters)	The actual name of the vendor.
└<homepageurl>	String (255 characters)	The website address for the vendor account. The protocol must be included for the url to be valid
└<supportemailaddress>	String (100 characters)	The support email address for the vendor account.
└<sendhtml emails>	String (3 characters)	Either YES if emails will be sent with HTML formatting, NO if not.
└<defaultterminalcurrency>	String (3 characters)	The default currency for terminal transactions. Must conform to the ISO 4217 3-letter currency code of a supported currency. If it doesn't conform, then the currency field is not updated

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>setVendorDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vendordetails>
    <vendorprovidedname>Online Cheese</vendorprovidedname>
    <homepage>www.yourbusinesshomepage.com</homepage>
    <supportemailaddress>support@onlinecheese.com</supportemailaddress>
    <sendhtml emails>YES</sendhtml emails>
    <defaultterminalcurrency>GBP</defaultterminalcurrency>
  </vendordetails>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.17 Admin Command - addBlockedCardRanges

This command adds card BIN ranges to the blocked card range list on the vendor account. By adding these BIN ranges, transactions will then be blocked from customers with cards within these ranges.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String addBlockedCardRanges	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedcardranges>	node	Node containing country code elements
L<blockedcardrange>	Numeric (9 characters)	The card BIN range to be blocked

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>addBlockedCardRanges</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedcardranges>
    <blockedcardrange>675967511</blockedcardrange>
  </blockedcardranges>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.18 Admin Command - addBlockedCountries

This command adds country codes to the blocked countries list on the vendor account. By adding these countries, transactions will then be blocked from customers within these countries.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String addBlockedCountries	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedcountries>	node	Node containing country code elements
^L <countrycode>	String (2 characters)	ISO3166 2 character country code

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>addBlockedCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedcountries>
    <countrycode>AF</countrycode>
    <countrycode>AL</countrycode>
  </blockedcountries>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.19 Admin Command - addBlockedIPs

This command adds selected blocked IP addresses and subnet masks to the blocked IP addresses list on the vendor account. By adding these IP addresses from the list, transactions will no longer be allowed from customers with these IP addresses.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String addBlockedIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedIPs>	node	Node containing the ipaddress nodes
L<ipaddress>	Node	Node containing the ipaddress and subnetmask values
L<address>	String (15 characters)	The IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
L<mask>	String (15 characters)	The subnet mask for the IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>addBlockedIPs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedips>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
    </ipaddress>
  </blockedips>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.20 Admin Command - addBlockedIssuingCountries

This command adds a country to the list of card issuing countries, these countries will be prevented from processing transactions against the provided vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String addBlockedIssuingCountries	The service being requested.
<vendor>	String (16 characters)	The vendor account against which this command will be executed.
<user>	String (20 characters)	Must be an admin user under the provided vendor
<blockedissuingcountries>	Node	
L<countrycode>	String (2 characters)	Must be an ISO 2 character country code.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>addBlockedIssuingCountries</command>
  <vendor>egoo</vendor>
  <user>egoo</user>
  <blockedissuingcountries>
    <countrycode>AF</countrycode>
    <countrycode>AL</countrycode>
  </blockedissuingcountries>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.21 Admin Command - addValidIPs

This command will add requested valid IP addresses and subnet masks to the valid IP addresses list on the vendor account. This list of valid IP addresses is used to validate where the transaction is being created from for Server and Direct transactions.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String addValidIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<validIPs>	node	Node containing the ipaddress nodes
L<ipaddress>	Node	Node containing the ipaddress and subnetmask values
L<address>	String (15 characters)	The IP address to be allowed in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
L<mask>	String (15 characters)	The subnet mask for the IP address to be allowed in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
L<note>	String (255 characters)	A description of the ip address

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>addValidIPs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <validips>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
      <note>my ip address</note>
    </ipaddress>
  </validips>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.22 Admin Command - deleteAVSCV2Rules

This command removes all existing AVS and CV2 rules on the vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deleteAVSCV2Rules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteAVSCV2Rules</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```


2.23 Admin Command - delete3DSecureRules

This command removes all existing 3DSecure rules on the vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String delete3DSecureRules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>delete3DSecureRules</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <timestamp>20/10/2010 15:59:55</timestamp>
</vspaccess>
```

2.24 Admin Command - deleteBlockedCardRanges

This command deletes card BIN ranges from the blocked card range list on the vendor account. By removing these BIN ranges, transactions will be allowed from customers with cards within these ranges.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deleteBlockedCardRanges	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedcardranges>	node	Node containing country code elements
L<blockedcardrange>	Numeric (9 characters)	The card BIN range to be blocked

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteBlockedCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedcardranges>
    <blockedcardrange>675967511</blockedcardrange>
    <blockedcardrange>688968686</blockedcardrange>
  </blockedcardranges>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
</vspaccess>
```

2.25 Admin Command - deleteBlockedCountries

This command removes country codes which are currently in the blocked countries list on the vendor account. By deleting these countries, transactions will then be allowed from customers within these countries.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deleteBlockedCountries	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedcountries>	node	Node containing country code elements
^L <countrycode>	String (2 characters)	ISO3166 2 character country code

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteBlockedCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedcountries>
    <countrycode>UK</countrycode>
  </blockedcountries>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.26 Admin Command - deleteBlockedIPs

This command deletes requested blocked IP addresses and subnet masks from the blocked IP addresses list on the vendor account. By deleting these IP addresses from the list, transactions will be allowed from customers with these IP addresses.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deleteBlockedIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedIPs>	node	Node containing the ipaddress nodes
L<ipaddress>	Node	Node containing the ipaddress and subnetmask values
L<address>	String (15 characters)	The IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
L<mask>	String (15 characters)	The subnet mask for the IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteBlockedIPs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <blockedips>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
    </ipaddress>
  </blockedips>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <timestamp>21/10/2010 09:40:12</timestamp>
</vspaccess>
```

2.27 Admin Command - deleteBlockedIssuingCountries

Removes a range of countries from the list of card issuing countries which are prevented from processing transactions against this vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String	The service being requested.
<vendor>	String (16 characters)	The vendor account against which this command will be executed.
<user>	String (20 characters)	Must be an admin user under the provided vendor
<blockedissuingcountries>	Node	
L<countrycode>	String (2 characters)	Must be an ISO 2 character country code.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteBlockedIssuingCountries</command>
  <vendor>egoo</vendor>
  <user>egoo</user>
  <blockedissuingcountries>
    <countrycode>AZ</countrycode>
  </blockedissuingcountries>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <timestamp>12:34:08 12-12-2010</timestamp>
</vspaccess>
```

2.28 Admin Command - deleteValidIPs

This command removes selected valid IP addresses and subnet masks which are currently in the valid IP addresses list on the vendor account. This list of valid IP addresses is used to validate where the transaction is being created from for Server and Direct transactions.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deleteValidIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<blockedIPs>	node	Node containing the ipaddress nodes
L<ipaddress>	Node	Node containing the ipaddress and subnetmask values
L<address>	String (15 characters)	The IP address to be allowed in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
L<mask>	String (15 characters)	The subnet mask for the IP address to be allowed in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteValidIPs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <validips>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
    </ipaddress>
  </validips>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <timestamp>21/10/2010 09:40:12</timestamp>
</vspaccess>
```

2.29 Admin Command - setPaymentPageTemplates

This command selects the type of payment page you would like to display to your customer. This is only available to merchants processing through Form or Server.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String setPaymentPageTemplates	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<templatetype>	String (9 characters)	<p>The type of payment page template you would like displaying to your customers:</p> <p>Default the standard Sage Pay templates with editable address field</p> <p>DefaultAddrRDOnly the standard Sage Pay templates with read-only address fields</p> <p>DefaultNoAddr the standard Sage Pay templates with no visible address fields</p> <p>Custom your own custom template design (previously submitted to sage Pay by you)</p>

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>
  <command>deleteBlockedIPs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <templatetype>default</templatetype>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <timestamp>20/10/2010 15:47:24</timestamp>
</vspaccess>
```

2.30 Admin Command - setPaypalAccount

The setPaypalAccount command sets the requested Paypal account. It is a two step process involving contacting Paypal to verify the email address exists.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String setPaypalAccount	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<paypaluserid>	String (100 characters)	The paypal account to be set

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>  
  <command>setPaypalAccount</command>  
  <vendor>sagepay</vendor>  
  <user>sagepay</user>  
  <paypaluserid>testmessage@sagepay.com</paypaluserid>  
</vspaccess>
```

Sample XML Output

```
<vspaccess>  
  <errorcode>0000</errorcode>  
  <timestamp>20/10/2010 15:59:55</timestamp>  
</vspaccess>
```


2.31 Admin Command - deletePaypalAccount

The deletePaypalAccount command deletes the requested Paypal account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String deletePaypalAccount	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.

Sample XML Input

```
<vspaccess>  
  <command>deletePaypalAccount</command>  
  <vendor>sagepay</vendor>  
  <user>sagepay</user>  
</vspaccess>
```

Sample XML Output

```
<vspaccess>  
  <errorcode>0000</errorcode>  
</vspaccess>
```

2.32 Reporting Command - doesUserExist

The doesUserExist command returns whether the user exists on the account of the provided Vendor name.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String doesUserExist	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<username>	Alphanumeric (20 characters)	The user name. Can only contain letters and numbers.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<vendorid>	String (40 characters)	The VendorId of the User (same as the vendor account)
<username>	String (20 characters)	The user's username.
<userexists>	String (3 characters)	The userexist setting of either YES or NO

Sample XML Input

```
<vspaccess>
  <command>doesUserExist</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <username>sagepay</username>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <vendorid>8117B0C2-899A-F3B5-2A3B-67A716DE3BBE</vendorid>
  <username>protx</username>
  <userexists>YES</userexists>
  <timestamp>20/10/2010 16:05:38</timestamp>
</vspaccess>
```

2.33 Reporting Command - getUserRights

This command returns a list of access privileges set on the provided 'username' detailing their access to the vendor's account and transaction information.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getUserRights	The service being requested.
<vendor>	String (16 characters)	The vendor account on which the user is present.
<user>	String (20 characters)	Must be an admin user on the vendor account
<username>	String (20 characters)	The user who's access rights will be returned.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<vendorid>	String (40 characters)	The vendorID of the vendor account the User is associated with.
<username>	String (20 characters)	The user who's access rights will be returned.
<admin>	String (3 characters)	YES or NO
<islocked>	String (3 characters)	YES or NO
<reports>	String (3 characters)	YES or NO
<terminal>	String (3 characters)	YES or NO
<transactions>	String (3 characters)	YES or NO
<updates>	String (3 characters)	YES or NO
<canabort>	String (3 characters)	YES or NO
<canmanual>	String (3 characters)	YES or NO
<canrefund>	String (3 characters)	YES or NO
<canrelease>	String (3 characters)	YES or NO
<canrepeat>	String (3 characters)	YES or NO
<canviewall>	String (3 characters)	YES or NO

<canvoid>	String (3 characters)	YES or NO
<loggedin>	String (3 characters)	YES or NO
<homepage>	String (20 characters)	The page which the user will initially be logged into

Sample XML Input

```
<vspaccess>
  <command>getUserRights</command>
  <vendor>egoo</vendor>
  <user>egoo</user>
  <username>John</username>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <vendorid>8117BOC2-899A-F3B5-2A3B-67A716DE3BBE</vendorid>
  <username>rob</username>
  <admin>YES</admin>
  <islocked>NO</islocked>
  <reports>YES</reports>
  <terminal>YES</terminal>
  <transactions>YES</transactions>
  <updates>YES</updates>
  <canabort>YES</canabort>
  <canmanual>YES</canmanual>
  <canrefund>YES</canrefund>
  <canrelease>YES</canrelease>
  <canrepeat>YES</canrepeat>
  <canviewall>YES</canviewall>
  <canvoid>YES</canvoid>
  <loggedin>YES</loggedin>
  <homepage>search.msp</homepage>
  <timestamp>20/10/2010 12:58:40</timestamp>
</vspaccess>
```

2.34 Reporting Command - getAccountLogs

Using this command option will retrieve the account log detailing all the changes to the Reporting database.

Request

Node/Element	Type	Description
<command>	String getAccountLogs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<logentries>	Node	Node containing all log entries
L<logrow>	Node	Node containing individual log entry
L<logdate>	Datetime DD/MM/YYYY HH:MM:SS:sss	The date and time of the log entry
L<logentry>	String (300 characters)	The log details

Sample XML Input

```
<vspaccess>
  <command>getAccountLogs</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <logentries>
    <logrow>
      <logdate>08/07/2010 15:42:22.207</logdate>
      <logentry>IP added by barry</logentry>
    </logrow>
  </logentries>
</vspaccess>
```

2.35 Reporting Command - getVendorUserList

This command will obtain a list of all user accounts on the account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String	The service being requested.
<vendor>	String (16 characters)	The Vendor Name
<user>	String (20 characters)	Must be an admin user under the Sage Pay account

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<users>	node	The node containing the returned users.
L <user>	node	The node representing the User.
L <username>	String (20 characters)	The user's username.
L <islocked>	String (Boolean)	Will contain either true or false to indicate whether the user account is locked from failed login attempts
L <lastactivity>	Datetime (dd/mm/yyyy hh:mm:ss)	A Timestamp indicating the last action of the user within the Sage Pay systems if they have an open session.
L <admin>	String (Boolean)	Will contain either true or false to indicate whether the user account is given Admin privileges in MSP.
L <transactions>	String (Boolean)	True or False dependant on the user's access right for the transaction section.
L <reports>	String (Boolean)	True or False dependant on the user's access right for the reporting section.
L <terminal>	String (Boolean)	True or False dependant on the user's access right for the terminal section.
L <updates>	String (Boolean)	True or False dependant on the user's access right for the updates section.
L <canviewall>	String (Boolean)	True or False dependant on the user's access right to view all types of transaction in My Sage Pay
L <canrefund>	String (Boolean)	True or False dependant on the user's access right to process refunds of transactions in My Sage Pay
L <canrelease>	String (Boolean)	True or False dependant on the user's access right to process releases of transactions in My Sage Pay
L <canrepeat>	String (Boolean)	True or False dependant on the user's access right to process repeats of transactions in My Sage Pay
L <canabort>	String (Boolean)	True or False dependant on the user's access right to process aborts of transactions in My Sage Pay
L <canvoid>	String (Boolean)	True or False dependant on the user's access right to process voiding of transactions in My Sage Pay

L<canmanual>	String (Boolean)	True or False dependant on the user's access right to process manual transactions in My Sage Pay
L<vendorid>	String (40 characters)	The VendorId of the User (same as the vendor account)
L<homepage>	String (30 characters)	The landing page which the user will be presented with when logging into My Sage Pay.

Sample XML Input

```
<vspaccess>
  <command>getVendorDetails</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <users>
    <user>
      <username>AnotherTest</username>
      <islocked>true</islocked>
      <lastactivity>20/10/2010 15:51:12.927</lastactivity>
      <admin>true</admin>
      <transactions>true</transactions>
      <reports>false</reports>
      <terminal>true</terminal>
      <updates>false</updates>
      <canviewwall>false</canviewwall>
      <canrefund>false</canrefund>
      <canrelease>false</canrelease>
      <canrepeat>false</canrepeat>
      <canabort>false</canabort>
      <canvoid>false</canvoid>
      <canmanual>false</canmanual>
      <vendorid>8117B0C2-899A-F3B5-2A3B-67A716DE3BBE</vendorid>
      <homepage>users.msp</homepage>
    </user>
  </users>
</vspaccess>
```

2.36 Reporting Command - getVendorDetails

Using this command will obtain the details of the provided vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String	The service being requested.
<vendor>	String (16 characters)	The Vendor Name
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<vendorprovidedname>	String (50 characters)	The vendors provided name.
<homepageurl>	String (255 characters)	The vendors website URL.
<fromemail>	String (100 characters)	The vendors registered email address for contact by customers and sagepay support.
<sendhtmlmails>	String (3 characters)	Will contain either YES or NO to indicate whether HTML emails are being sent.
<defaultterminalcurrency>	String (3 characters)	The default transaction currency for the provided vendor as an ISO 4217 currency code.
<logo>	String (255Characters)	The filename name of the current logo on the account
<closed>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time of when the account was closed
<isgiftaid>	String (3 characters)	Will contain either YES or NO to indicate the vendor account is set up for giftaid
<isavscv2>	String (3 characters)	Will contain either YES or NO to indicate the vendor account is set up for avscv2
<isthreed>	String (3 characters)	Will contain either YES or NO to indicate the vendor account is set up for Three D Secure
<terminalonly>	String (3 characters)	Will contain either YES or NO to indicate the vendor account is terminal only
<directrefund>	String (3 characters)	Will contain either YES or NO to indicate the vendor account has direct refund
<paypaluserid>	String (100 characters)	The paypal user id.

<code><templatetype></code>	String (9 characters)	<p>The type of payment page template currently displayed to your customers:</p> <p>default the standard Sage Pay templates with editable address field</p> <p>readonly the standard Sage Pay templates with read-only address fields</p> <p>noaddress the standard Sage Pay templates with no visible address fields</p> <p>custom your own custom template design (previously submitted to sage Pay by you)</p>
-----------------------------------	--------------------------	--

Sample XML Input

```
<vspaccess>
  <command>getVendorDetails</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <vendorprovidedname>egoo</vendorprovidedname>
  <homepageurl>www.extragoo.com</homepageurl>
  <fromemail>info@extragoo.com</fromemail>
  <sendhtmlemails>YES</sendhtmlemails>
  <defaultterminalcurrency>GBP</defaultterminalcurrency>
  <logo>mylogo.gif</logo>
  <isclosed></isclosed>
  <isgiftaid>NO</isgiftaid>
  <isavscv2>NO</isavscv2>
  <isthreed>NO</isthreed>
  <terminalonly>NO</terminalonly>
  <directrefund>NO</directrefund>
  <paypaluserid>bob@impactpt.co.uk</paypaluserid>
  <templatetype>noaddress</templatetype>
</vspaccess>
```

2.37 Reporting Command - getVendorPaymentSystems

Using this command will obtain the details of the payment systems available to the provided vendor account.

Request

Node/Element	Type	Description
<command>	String getVendorPaymentSystems	The service being requested.
<vendor>	String (16 characters)	The Vendor Name
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command
<currency> (optional)	String (3 characters)	ISO 3 character currency code. eg: GBP
<accounttype> (optional)	String (1 character)	The account type of the payment system. E,C or M.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<paymentsystems>	Node	The vendors provided name.
L<paymentsystem>	String (50 characters)	The name of the payment system (card type) acceptable to this vendor account.

Sample XML Input

```
<vspaccess>
  <command>getVendorDetails</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
  <currency>GBP</currency>
  <accounttype>M</accounttype>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <paymentsystems>
    <paymentsystem>American Express</paymentsystem>
    <paymentsystem>MasterCard</paymentsystem>
    <paymentsystem>Solo</paymentsystem>
    <paymentsystem>UK Maestro</paymentsystem>
    <paymentsystem>Visa</paymentsystem>
    <paymentsystem>Visa Debit / Delta</paymentsystem>
    <paymentsystem>Visa Electron</paymentsystem>
  </paymentsystems>
</vspaccess>
```

2.38 Reporting Command - getValidIPs

The getValidIPs command gets a list of the currently valid IPs that are set up on the account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getValidIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<ipaddresses>	Node	Node containing the ipaddress nodes
<ipaddress>	Node	Node containing the ipaddress and subnetmask values
<address>	String (15 characters)	The IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
<mask>	String (15 characters)	The subnet mask for the IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
<note>	String (255 characters)	A description of the ip address

Sample XML Input

```
<vspaccess>
  <command>getValidIPs</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <ipaddresses>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
      <note>API Added IP</note>
    </ipaddress>
  </ipaddresses>
</vspaccess>
```

2.39 Reporting Command - getURLs

This command will return the fully qualified service URLs for the transaction types supported by the vendor account. By default, all URLs for all supported transaction types are returned. You can optionally specify the transaction types you wish to receive URLs to, if you wish to filter the returned list.



This command doesn't require the user to have administrator rights.

See also: `getTransactionTypes`

Request

Node/Element	Type	Description
<command>	String getURLs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<system>	String (10 characters)	The system for which the URLs are being requested. Must be either VSPServer , VSPDirect or VSPForm
<txtypes> (optional)	Node	Container for the transaction types being returned
L<txtype> (optional)	String (20 characters)	The transaction types supported. Contains PAYMENT, REFUND, DEFERRED, RELEASE, ABORT, REPEAT, REPEATDEFERRED, VOID, MANUAL, AUTHENTICATE, CANCEL or AUTHORISE

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<urls>	Node	Container for the URLs being returned
L<url>	Node	Container for a single URL with the service it matches
L<txtype>	String (20 characters)	The name of the transaction type for which this URL maps
L<serviceURL>	String - FQDN (255 characters)	Fully Qualified Domain Name URL to the service named above.

Example request on overleaf..

Sample Input XML:

```
<vspaccess>
  <command>getURLs</command>
  <vendor>egoo</vendor>
  <user>sagepay</user>
  <system>VSPDirect</system>
  <txtypes>
    <txtype>PAYMENT</txtype>
    <txtype>REFUND</txtype>
    <txtype>VOID</txtype>
  </txtypes>
  <signature>6AB000A9914E874CF3937D12F11ABD1B</signature>
</vspaccess>
```

Sample Response XML:

```
<vspaccess>
  <errorcode>0000</errorcode>
  <urls>
    <url>
      <txtype>PAYMENT</txtype>
    </url>
    <url>
      <txtype>REFUND</txtype>
    </url>
  </urls>
</vspaccess>
```

2.40 Reporting Command - get3DSecureStatus

This command allows you to switch 3D Secure checks on or off for the chosen vendor. You cannot remove or add rules with this command, so switching on 3D Secure will also turn on existing 3D Secure rules (if they exist on this account)



The user name provided must have administrative privileges to run this command.

See also: getAVSCV2Status, setAVSCV2Status, get3DSecureStatus

Request

Node/Element	Type	Description
<command>	String set3DSecureStatus	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<status>	String (3 characters)	Should contain either ON or OFF to indicate the preferred status of the 3D-Secure checks.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<status>	String (3 characters)	Contains either ON or OFF to indicate whether 3DSecure checks are on or off.
<rules>	String (3 characters)	Contains either ON or OFF to indicate whether a rulebase is applied to the 3DSecure checks.

Sample XML input:

```
<vspaccess>  
  <command>set3DSecureStatus</command>  
  <vendor>sagepay</vendor>  
  <user>charlie</user>  
  <status>ON</status>  
</vspaccess>
```

Sample Return XML:

```
<errorcode>0000</errorcode>  
<status>ON</status>  
<rules>OFF</rules>
```

2.41 Reporting Command - get3DSecureRules

The get3DSecureRules command gets a list of the 3D Secure Rules on your account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String get3DSecureRules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<rules>		
<rule>	Node	Node containing the ipaddress nodes
<startvalue>	String (15 characters)	The starting value you wish the particular rule to apply from (must not overlap existing rules)
<endvalue>	String (15 characters)	The finishing value you wish the particular rule to apply to (must not overlap existing rules)
<perform3dauth>	String (3 characters)	YES or NO
<allownon3dcards>	String (3 characters)	YES or NO
<allownon3dissuers>	String (3 characters)	YES or NO
<allow3dfailures>	String (3 characters)	YES or NO
<allowmpiauths>	String (3 characters)	YES or NO

Sample XML Input

```
<vspaccess>
  <command>get3DSecureRules</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <rules>
    <rule>
      <startvalue>0.00</startvalue>
      <endvalue>1000.00</endvalue>
      <perform3dauth>YES</perform3dauth>
      <allownon3dcards>YES</allownon3dcards>
      <allownon3dissuers>YES</allownon3dissuers>
      <allow3dfailures>YES</allow3dfailures>
      <allowmpiauths>YES</allowmpiauths>
    </rule>
  </rules>
  <timestamp>21/10/2010 09:40:12</timestamp>
</vspaccess>
```

2.42 Reporting Command - getAVSCV2Status

The getAVSCV2Status command returns information about whether AVS/CV2 fraud screening is on or off, and whether rules have been applied.



The user name provided must have administrative privileges to run this command.

See also: get3DSecureStatus, setAVSCV2Status, set3DSecureStatus

Request

Node/Element	Type	Description
<command>	String getAVSCV2Status	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<status>	String (3 characters)	Contains either ON or OFF to indicate whether AVS/CV2 checks are on or off.
<rules>	String (3 characters)	Contains either ON or OFF to indicate whether a rulebase is applied to the AVS/CV2 checks.

Sample XML Input

```
<vspaccess>
  <command>getAVSCV2Status</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <status>ON</status>
  <rules>ON</rules>
</vspaccess>
```


2.43 Reporting Command - getAvsCv2Rules

This command returns the rule set for the AVS/CV2 authentication system.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getAVSCV2Rules	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<rules>	node	Node containing rules
L<rule>	node	Container for the rule
L<startvalue>	Numeric (monitary value)	The starting value you wish the particular rule to apply from (must not overlap existing rules)
L<endvalue>	Numeric (monitary value)	The finishing value you wish the particular rule to apply to (must not overlap existing rules)
L<allownodatamatches>	String (3 characters)	YES or NO
L<allowaddressmatchonly>	String (3 characters)	YES or NO
L<allowsecuritycodematchonly>	String (3 characters)	YES or NO
L<allowdatanotchecked>	String (3 characters)	YES or NO

Sample XML Input

```
<vspaccess>
  <command>getAVSCV2Rules</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Example response overleaf..

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <rules>
    <rule>
      <startvalue>1.00</startvalue>
      <endvalue>100.00</endvalue>
      <allownodatamatches>NO</allownodatamatches>
      <allowaddressmatchonly>YES</allowaddressmatchonly>
      <allowsecuritycodematchonly>NO</allowsecuritycodematchonly>
      <allowdatanotchecked>NO</allowdatanotchecked>
    </rule>
  </rules>
  <timestamp>20/10/2010 11:57:39</timestamp>
</vspaccess>
```

2.44 Reporting Command - getBatchDetail

This command returns a list of all transactions sent to the bank in the specified settlement file (batch). This command is equivalent to the mySagePay Batch Detail Report (but unlike the report, transactions are not split into payments and refunds.)



This command doesn't require the user to have administrator rights.

See also: getTransactionSummary, getTransactionDetail, getBatchList

NOTE: This command returns a maximum of 50 rows of information. The `<startrow>` and `<endrow>` fields show which rows have been returned, with `<totalrows>` showing the total amount of information available.

You can make a request to return specific subsets of the information available by setting the `<startrow>` and `<endrow>` values in your request. So for example, if there are 140 rows of information in the batch, your first request will return rows 1 to 50. You can make two subsequent requests to retrieve rows 51 to 100, and 101 to 140 by setting the start and end rows accordingly.

Request

Node/Element	Type	Description
<code><command></code>	String getBatchDetail	The service being requested.
<code><vendor></code>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<code><user></code>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<code><batchid></code>	Numeric (Integer)	The unique ID number of this batch for this acquirer
<code><authprocessor></code>	String (50 characters)	The name of the acquirer to whom this batch was sent
<code><startrow></code> (optional)	Integer	The first row of information you require. If left blank this defaults to the first row.
<code><endrow></code> (optional – must be less than 50 if the startrow is not provided)	Integer	The last row of information you require. By default this is set to 50. A smaller window can be specified.

Response overleaf

Response (continued overleaf)

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<transactions>	Node	A container for the list of transactions returned
L<totalrows>	Integer	The total number of rows present.
L<transaction>	Node	A container for the individual transaction
L<rownumber>	Integer	The current row of information returned.
L<startrow>	Integer	The first row of information returned.
L<endrow>	Integer	The last row of information returned.
L<vpstxid>	String (40 characters)	The unique VPSTxid (transaction id) of this transaction
L<vendortxcode>	String (40 characters)	The unique VendorTxCode of this transaction
L<transactiontype>	String (15 characters)	The type of transaction. Values are: Authenticate, Authorise, Deferred, Manual, Payment, PreAuth, Refund, Repeat and RepeatDeferred
L<systemused>	String (1 character)	Contains F for Form, S for Server, D for Direct or T for Terminal.
L<amount>	Numeric (monetary value)	The value of the transaction
L<currency>	String (3 characters)	The ISO 4217 currency code of this transaction
L<started>	Datetime (dd/mm/yyyy hh:mm:ss.SSS)	The date/time that the transaction was registered with Sage Pay
L<vpsauthcode> (optional)	Numeric (big integer value)	The unique Sage Pay auth code for a successful transaction only
L<bankauthcode> (optional)	String (20 characters)	The authorisation code from the bank. For PayPal transactions this is the PayPal transaction ID
L<cv2result> (optional)	String (12 characters)	The results of the CV2 fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<addressresult> (optional)	String (12 characters)	The results of the AVS address fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<postcoderesult> (optional)	String (12 characters)	The results of the AVS postcode fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<threedresult> (optional)	String (12 characters)	The results of the 3D-Secure fraud checks. Either OK , NOTCHECKED , ATTEMPTONLY , NOTAVAILABLE , NOTAUTHED , INCOMPLETE or ERROR
L<cardholder> (optional)	String (100 characters)	If available, the name on the credit/debit card.
L<released> (optional)	Datetime	The date/time this transaction was released, if applicable
L<refunded> (optional)	String (8 characters)	Either YES if fully refunded, NO if not refunded or PARTIAL if partly refunded
L<paymentsystem> (optional)	String (20 characters)	The payment system used for the transaction

L<repeated> (optional)	String (3 characters)	Either YES if at least one REPEAT has occurred against this transaction, NO if none have
L<username> (optional)	String (20 characters)	If available, the name of the user account in MySagePay who processed this transaction.
L<t3mscore> (optional)	Numeric	A value in the range -100 to +100 as a fraud score from the third man, when available.
L<t3maction> (optional)	String (10 characters)	Either OK , HOLD , REJECT or NORESULT as a recommended course of action based on the fraud score.
L<t3mid> (optional)	String (20 characters)	The ID of this transaction in the 3 rd Man systems.

Sample XML Input

```
<vspaccess>
  <command>getBatchDetail</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <batchid>123456789</batchid>
  <authprocessor>SomeAcquirerXyz</authprocessor>
  <startrow>1</startrow>
  <endrow>50</endrow>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <transactions>
    <transaction>
      <rownumber>7</rownumber>
      <startrow>1</startrow>
      <endrow>7</endrow>
      <vpstxid>987A9292-409F-4F23-A4C1-861A78F7C340</vpstxid>
      <systemused>T</systemused>
      <transactiontype>Payment</transactiontype>
      <vendortxcode>T-protx-100524170853-114</vendortxcode>
      <started>24/05/2010 17:09:30.333</started>
      <amount>23.50</amount>
      <currency>GBP</currency>
      <vpsauthcode>269956638</vpsauthcode>
      <bankauthcode>024576</bankauthcode>
      <addressresult>MATCHED</addressresult>
      <postcoderesult>MATCHED</postcoderesult>
      <cv2result>MATCHED</cv2result>
      <cardholder>Greg Cahill V0016487</cardholder>
      <refunded>NO</refunded>
      <repeated>NO</repeated>
      <released/>
      <username>darren</username>
      <t3mscore>-30</t3mscore>
      <t3maction>OK</t3maction>
      <t3mid>559442606</t3mid>
      <threedresult>NOTCHECKED</threedresult>
      <paymentsystem>Visa Debit / Delta</paymentsystem>
    </transaction>
  </transactions>
  <totalrows>7</totalrows>
</vspaccess>
<signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.45 Reporting Command - getBatchList

This command returns a list of all bank settlement (batch) files scheduled between the specific dates for the given vendor (equivalent to the Batch Reports in My Sage Pay).



This command doesn't require the user to have administrator rights.

Note: The list is limited to 1 month periods to minimise impact on the database. Multiple queries would be needed to return results for longer periods.

See also: getTransactionList, getTransactionDetail, getBatchDetail

Request

Node/Element	Type	Description
<command>	String getBatchList	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startdate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time from which to begin searching for transactions
<enddate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time of the last transaction to be returned. Cannot be more than a month after the start date

Response (continued overleaf)

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<batches>	Node	Container for the batch information
L<batch>	Node	Container for single batch file details
L<authprocessor>	String (50 characters)	The name of the acquirer to whom this batch was sent
L<batchid>	Numeric (Integer)	The unique ID number of this batch for this acquirer
L<completed>	Datetime (dd/mm/yyyy hh:mm:ss.SSS)	The date and time, including milliseconds, the file was submitted to the acquiring bank.
L<transactiongroups>	Node	Container for the transaction summary information
L<transactiongroup>	Node	Container for the transaction summary information per currency

Response (continued)

<code>L<currency></code>	String (3 characters)	Contains the ISO 4217 3-letter currency code of the batch currency.
<code>L<paymentnumber></code>	Numeric (Integer)	The number of payment transactions in this batch for this currency
<code>L<paymentvalue></code>	Numeric (Monetary value)	The total value of the payment transactions in this batch for this currency
<code>L<refundnumber></code>	Numeric (Integer)	The number of refund transactions in this batch for this currency
<code>L<refundvalue></code>	Numeric (Monetary value)	The total value of the refund transactions in this batch for this currency

Sample XML Input

```
<vspaccess>
  <command>getBatchList</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <startdate>01/01/2010 00:00:00</startdate>
  <enddate>02/01/2010 00:00:00</enddate>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <batches>
    <batch>
      <authprocessor>Elavon</authprocessor>
      <batchid>1234</batchid>
      <completed>01/01/2010 12:00:00</completed>
      <transactiongroups>
        <transactiongroup>
          <currency>GBP</currency>
          <paymentnumber>1</paymentnumber>
          <paymentvalue>10.00</paymentvalue>
          <refundnumber>0</refundnumber>
          <refundvalue>0.00</refundvalue>
        </transactiongroup>
      </transactiongroups>
    </batch>
  </batches>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.46 Reporting Command - getCardType

This command returns the and the type of card from the first 1-9 digits supplied, allowing you to look up a card type for validation purposes, or correct submission to Direct.



This command doesn't require the user to have administrator rights.

Request

Node/Element	Type	Description
<command>	String getCardType	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<cardbegins>	Numeric (1-9 characters)	The first 1-9 digits of the card number. 9 gives the most accurate results. No spaces or separators must be included. Numbers only

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<paymentsystem>	String (12 characters)	The card type used in this transaction. Contains either AMEX , DC (Diner's Club), DELTA (Visa Debit), JCB , MAESTRO , MC (MasterCard), PAYPAL , SOLO , SWITCH (UK Maestro), UKE (Visa Electron), VISA (Visa Credit) or UNKNOWN if card range not recognised

Sample XML Input

```
<vspaccess>
  <command>getCardType</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <cardbegins>123456789</cardbegins>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <paymentsystem>test</paymentsystem>
  <errorcode>0000</errorcode>
  <error></error>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```


2.47 Reporting Command - getCardDetails

This command returns the card details from the first 1-9 digits supplied.



This command doesn't require the user to have administrator rights.

Request

Node/Element	Type	Description
<command>	String getCardDetails	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<cardbegins>	Numeric (1-9 characters)	The first 1-9 digits of the card number. 9 gives the most accurate results. No spaces or separators must be included. Numbers only

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<paymentsystemname>	String (20 characters)	The payment system (card type) name of the card being used. One of the following: American Express, Diners Club, JCB, LASER, Maestro, MasterCard, PayPal, Solo, UK Maestro, Visa, Visa Debit / Delta, Visa Electron.
<shortname>	String (10 characters)	The short name of the card type used in this transaction. Contains either AMEX , DC (Diner's Club), DELTA (Visa Debit), JCB , MAESTRO , MC (MasterCard), PAYPAL , SOLO , SWITCH (UK Maestro), UKE (Visa Electron), VISA (Visa Credit) or UNKNOWN if card range not recognised
<issuedigits>	1-digit numeric	The number of issue digits for the card
<minimum>	9-digit numeric	The minimum number of the bin range
<maximum>	9-digit numeric	The maximum number of the bin range
<countrycode>	String (2 characters)	Contains the ISO 2-letter country code of a country.
<issuer>	String (255 characters)	The card issuer
<corporatecard>	String (3 characters)	Will contain either YES or NO to indicate whether or not the card is a corporate card
<iscredit>	String (3 characters)	Will contain either YES or NO to indicate whether or not the card is a credit card

Example request on overleaf..

Sample XML Input

```
<vspaccess>
  <command>getCardType</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <cardbegins>540400000</cardbegins>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <paymentsystemname>MasterCard</paymentsystemname>
  <shortname>MC</shortname>
  <issuedigits>-1</issuedigits>
  <minimum>540400000</minimum>
  <maximum>540400999</maximum>
  <countrycode>RU</countrycode>
  <issuer>Joint Stock Commercial Bank "S"</issuer>
  <corporatecard>NO</corporatecard>
  <iscredit>YES</iscredit>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.48 Reporting Command - getCurrencies

This command returns a list of all currencies supported by the vendor account.



This command doesn't require the user to have administrator rights.

See also: getMerchantAccounts

Request

Node/Element	Type	Description
<command>	String getCurrencies	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<accounttypes> (optional)	Node	Container for the account types being requested
L<accounttype> (optional)	String (1 character)	The account types being requested. Either: E for eCommerce M for Mail Order/Telephone Order C for Continuous Authority If not provided all accounttypes will be searched

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<currencies>	Node	A container for the currency list
L<currency>	String (3 characters)	Contains the ISO 4217 3-letter currency code of a supported currency.

Sample XML Input

```
<vspaccess>
  <command>getCurrencies</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <accounttypes>
    <accounttype>E</accounttype>
  </accounttypes>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <currencies>
    <currency>GBP</currency>
  </currencies>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.49 Reporting Command - getMerchantAccounts

The getMerchantAccounts command returns a list of the merchant number and bank account details associated with the Sage Pay vendor account. Optionally the request message can be limited to specific currencies and account types, but by default all merchant numbers for each account type (e-commerce, MOTO and continuous authority), for each currency, for each card type will be returned.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getMerchantAccounts	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<currencies> (optional)	Node	A container for the currency list
L<currency>	String (3 characters)	Contains the ISO 4217 3-letter currency code of a supported currency.
<accounttypes> (optional)	Node	A container for account type list
L<accounttype>	String (1 character)	Contains either E, C or M to filter the list of returned accounts

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<merchantaccounts>	Node	A container for the merchant number list
L<merchantaccount>	Node	Container for this merchant account entry
L<merchantnumber>	String (20 characters)	The merchant number of this account
L<authprocessor>	String (50 characters)	The acquirer who issued the merchant number
L<accountnumber>	String (20 characters)	The bank account associated with this merchant number
L<sortcode>	String (10 characters)	The sort code for the bank account
L<accountcurrency>	String (3 characters)	Contains the ISO 4217 3-letter code of the settlement currency for the bank account
L<streamlinenumber> (optional)	String (15 characters)	The NatWest Streamline company number for this merchant account (ONLY if acquirer is Streamline)
L<currencies> (optional)	Node	A container for the currency list
L<currency>	String (3 characters)	Contains the ISO 4217 3-letter currency code of a supported currency.
L<accounttypes> (optional)	Node	A container for account type list
L<accounttype>	String (1 character)	Contains either E, C or M to indicate the accounts types supported on this merchant number.

<code><paymentsystems></code> (optional)	Node	A container for payment system list
<code><paymentsystem></code>	String (12 characters)	The card types supported by this account. Contains either AMEX, DC (Diner's Club), DELTA (Visa Debit), JCB, MAESTRO, MC (MasterCard), PAYPAL, SOLO, SWITCH (UK Maestro), UKE (Visa Electron) or VISA (Visa Credit)

Sample XML Input

```
<vspaccess>
  <command>getMerchantAccounts</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <currencies>
    <currency>GBP</currency>
    <currency>EUD</currency>
  </currencies>
  <accounttypes>
    <accounttype>E</accounttype>
    <accounttype>M</accounttype>
  </accounttypes>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <command>getURLs</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <merchantaccounts>
    <merchantaccount>
      <merchantnumber>2030405060708090</merchantnumber>
      <authprocessor>BARCLAYS MERCHANT SERVICES</merchantnumber>
      <accountnumber>1231515151</accountnumber>
      <sortcode>204060</sortcode>
      <accountcurrency>204060</accountcurrency>
      <currencies>
        <currency>GBP</currency>
        <currency>USD</currency>
      </currencies>
      <accounttypes>
        <accounttype>E</accounttype>
        <accounttype>M</accounttype>
      </accounttypes>
      <paymentsystems>
        <paymentsystem>AMEX</paymentsystem>
        <paymentsystem>MC</paymentsystem>
        <paymentsystem>VISA</paymentsystem>
      </paymentsystems>
    </merchantaccount>
  </merchantaccounts>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.50 Reporting Command - getTransactionCardDetails

This command returns the last 4 digits of the card number, and the type of card used in the specified successful transaction. Only information about a transaction associated with the vendor can be returned. Transactions can be specified by either VendorTxCode or VPSTxID.

This is a subset of the getTransactionDetails request, to provide often requested functionality.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

See also: getTransactionSummary, getTransactionList, getBatchDetail

Request

Node/Element	Type	Description
<command>	String getTransactionCardDetails	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<vendortxcode> (optional) OR	String (40 characters)	The VendorTxCode of the transaction
<vpstxid> (optional)	String (40 characters)	The VPSTxID (transactionid) of the transaction

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<paymentsystem>	String (12 characters)	The card type used in this transaction. Contains either AMEX , DC (Diner's Club), DELTA (Visa Debit), JCB , MAESTRO , MC (MasterCard), PAYPAL , SOLO , SWITCH (UK Maestro), UKE (Visa Electron) or VISA (Visa Credit)
<last4digits>	String (4 characters)	The last 4 digits of the card number, if available

Example request on overleaf..

Sample XML Input

```
<vspaccess>
  <command>getTransactionCardDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vendortxcode>01Jan2010Transaction12345</vendortxcode>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Or

```
<vspaccess>
  <command>getTransactionCardDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vpstxid>00AA0A0A-0AA0-00A0-A000-A0000A0AA000</vpstxid>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <paymentsystem>VISA</paymentsystem>
  <last4digits>0000</last4digits>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.51 Reporting Command - getTransactionDetail

This command returns all information held in the database about the specified transaction. Only a transaction associated with the vendor can be returned. Transactions can be specified by either VendorTxCode or VPSTxID.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

See also: [getTransactionSummary](#), [getTransactionList](#), [getBatchDetail](#)

Request

Node/Element	Type	Description
<command>	String getTransactionDetail	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<vendortxcode> (optional) OR	String (40 characters)	The VendorTxCode of the transaction
<vpstxid> (optional)	String (40 characters)	The VPSTxID (transactionid) of the transaction

Response (continued overleaf)

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<vpstxid>	String (40 characters)	The unique VPSTxID (transaction id) of this transaction
<vendortxcode>	String (40 characters)	The unique VendorTxCode of this transaction
<transactiontype>	String (15 characters)	The type of transaction. Values are: Authenticate, Authorise, Deferred, Manual, Payment, PreAuth, Refund, Repeat and RepeatDeferred
<txstateid>	Integer	Identifier representing the status of the transaction
<status>	String (255 characters)	A human readable description of the transaction status.
<relatedtransactionid> (optional)	String (40 characters)	If this transaction is related to another e.g. a REFUND would be related to a parent PAYMENT, this is the vpstxid of that parent
<relatedvendortxcode> (optional)	String (40 characters)	The VendorTxCode of the related transaction.
<relatedamount> (optional)	Numeric (monetary value)	The Amount of the related transaction
<relatedcurrency> (optional)	String (3 characters)	The ISO 4217 currency code of the related transaction

Response (continued overleaf)

Response (continued)

<relatedstarted> (optional)	Datetime	The date/time that the related transaction was started.
<description>	String (200 characters)	The description of good passed at transaction registration
<amount>	Numeric (monetary value)	The value of the transaction
<currency>	String (3 characters)	The ISO 4217 currency code of this transaction
<started>	Datetime	The date/time that the transaction was registered with Sage Pay
<completed>	Datetime	The date/time that the transaction was completed
<securitykey>	String (10 characters)	The key used for signing the transaction completion messages
<clientip> (optional)	String (15 characters)	The IP address of the customer associated with this transaction (where available)
<giftaid>	String (3 characters)	Either YES if GiftAid applies to this transaction, or NO if it does not
<paymentsystem> (optional)	String (12 characters)	The card type used in this transaction. Contains either AMEX , DC (Diner's Club), DELTA (Visa Debit), JCB , MAESTRO , MC (MasterCard), PAYPAL , SOLO , SWITCH (UK Maestro), UKE (Visa Electron) or VISA (Visa Credit)
<paymentsystemdetails> (optional)	String (100 characters)	More information about the card used, where available (Issuer, Country etc.)
<startdate> (optional)	String (4 characters)	The card start date in MMYY format, if available
<expirydate> (optional)	String (4 characters)	The card expiry date in MMYY format, if available
<last4digits> (optional)	String (4 characters)	The last 4 digits of the card number, if available
<authprocessor> (optional)	String (50 characters)	The name of the acquirer who processed the transaction
<merchantnumber> (optional)	String (20 characters)	The merchant number through which this transaction passed
<accounttype> (optional)	String (1 character)	Contains either E , C or M to show the account type used in this transaction
<vpsauthcode> (optional)	Numeric (big integer value)	The unique Sage Pay auth code for a successful transaction only
<bankauthcode> (optional)	String (20 characters)	The authorisation code from the bank. For PayPal transactions this is the PayPal transaction ID
<batchid> (optional)	Numeric (integer value)	If the transaction has been settled with the bank, this is the ID of that batch
<billingfirstnames> (optional)	String	The customer's firstnames on the bill.
<billingsurname> (optional)	String	The customer's surname on the bill.
<billingaddress> (optional)	String (500 characters)	The customer's first line of billing address (without the postcode) if available
<billingaddress2> (optional)	String (500 characters)	The customer's second line of billing address (without the postcode) if available
<billingcity> (optional)	String	The customer's billing city if available

Response (continued)

<billingstate> (optional)	String	The customer's billing state if available
<billingpostcode> (optional)	String (50 characters)	The customer's billing address post code if available
<billingcountry> (optional)	String	The customer's billing country
<billingphone> (optional)	String	The customer's billing phone if possible
<deliveryfirstnames> (optional)	String	The customer's name for delivery
<deliversurname> (optional)	String	The customer's surname for delivery
<deliveryaddress> (optional)	String (500 characters)	The customer's first line of delivery address (without the postcode) if available
<deliveryaddress2> (optional)	String (500 characters)	The customer's second line of delivery address (without the postcode) if available
<deliverycity> (optional)	String	The customer's city for delivery
<deliverystate> (optional)	String	The customer's state for delivery
<deliverypostcode> (optional)	String (50 characters)	The customer's delivery address post code if available
<deliverycountry> (optional)	String	The customer's country for delivery
<deliveryphone> (optional)	String	The customer's phone for delivery
<cardholder> (optional)	String (100 characters)	If available, the name on the credit/debit card.
<cardfirstnames> (optional)	String	The names of the card holder.
<cardsurname> (optional)	String	The surname of the card holder.
<cardaddress> (optional)	String (500 characters)	The customer's registered first line of card address (without the postcode) if available
<cardaddress2> (optional)	String (500 characters)	The customer's registered second line of card address (without the postcode) if available
<cardcity> (optional)	String	The customer's registered card address city if available
<cardstate> (optional)	String	The customer's registered card address state if available
<cardpostcode> (optional)	String (50 characters)	The customer's registered card address post code if available
<cardcountry> (optional)	String	The customer's registered card address country if available
<customername> (optional)	String (100 characters)	The customer's name, if available
<customeremail> (optional)	String (100 characters)	The customer's e-mail address if available
<contactnumber> (optional)	String (20 characters)	The customer's telephone number if available
<contactfax> (optional)	String (20 characters)	The customer's fax number if available
<vendoremail> (optional)	String (100 characters)	The vendor e-mail address passed with this transaction

Response (continued)

<systemused>	String (1 character)	Contains F for VSPForm, S for VSPServer, D for VSPDirect or T for VSPTerminal.
<vpsprotocol>	String	Current version of gateway protocol
<callbackurl> (optional)	String	Callback url to which notification posts are sent
<aborted> (optional)	Datetime	The date/time this transaction was aborted, if applicable
<released> (optional)	Datetime	The date/time this transaction was released, if applicable
<refunded> (optional)	String (8 characters)	Either YES if fully refunded, NO if not refunded or PARTIAL if partly refunded
<repeated> (optional)	String (3 characters)	Either YES if at least one REPEAT has occurred against this transaction, NO if none have
<username> (optional)	String (20 characters)	If available, the name of the user account in MySagePay who processed this transaction.
<basket> (optional)	String (7500 characters)	If available, the full basket contents associated with this transaction
<applyavscv2>	Numeric	Value 0 to 3. See the protocols for more information about these values
<apply3dsecure>	Numeric	Value 0 to 3. See the protocols for more information about these values
<authattempt> (optional)	Numeric	A value between 1 and 3 to indicate the number of customer attempts to gain and authorisation (if they proceeded that far)
<cv2result> (optional)	String (12 characters)	The results of the CV2 fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
<addressresult> (optional)	String (12 characters)	The results of the AVS address fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
<postcoderesult> (optional)	String (12 characters)	The results of the AVS postcode fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
<threedattempt>	Numeric	The number of customer attempts to gain 3D Secure authentication.
<threedresult> (optional)	String (12 characters)	The results of the 3D-Secure fraud checks. Either OK , NOTCHECKED , ATTEMPTONLY , NOTAVAILABLE , NOTAUTHED , INCOMPLETE or ERROR
<eci> (optional)	Numeric	The ECI code sent to bank during authorisation or 3D-Secure transactions
<cavv> (optional)	String (32 characters)	The encoded CAVV or UCAF value for this secured transaction, if available
<t3mscore> (optional)	Numeric	A value in the range -100 to +100 as a fraud score from the third man, when available.
<t3maction> (optional)	String (10 characters)	Either OK , HOLD or REJECT as a recommended course of action based on the fraud score.
<t3mid> (optional)	String (20 characters)	The ID of this transaction in the 3 rd Man systems.
<emailmessage> (optional)	String	
<invalidstatus> (optional)	String	

<invalidstatusdetail> (optional)	String	
<invalidrequestsource> (optional)	String	
<locale>	String	
<xid>	String	An authorisation code that is part of the 3D Secure process.

Sample XML Input

```
<vspaccess>
  <command>deleteUser</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vendortxcode>01Jan2010Transaction12345</vendortxcode>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Or:

```
<vspaccess>
  <command>deleteUser</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vpstxid>B623A93F-D955-5EB7-BF0C-13CCAD2FAEBC</vpstxid>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <vpstxid>B623A93F-D955-5EB7-BF0C-13CCAD2FAEBC</vpstxid>
  <vendortxcode>T-protx-100824022248-163</vendortxcode>
  <transactiontype>Payment</transactiontype>
  <txstateid>1</txstateid>
  <status>Transaction failed registration. Either an INVALID or MALFORMED response was
returned.</status>
  <relatedtransactionid/>
  <relatedvendortxcode/>
  <relatedamount>0.00</relatedamount>
  <relatedcurrency/>
  <relatedstarted/>
  <description>Terminal Payment taken by gbtest1</description>
  <amount>445.00</amount>
  <currency>GBP</currency>
  <started>24/08/2010 14:27:54.403</started>
  <completed>24/08/2010 14:27:54.360</completed>
  <securitykey>WCKS7OK1EM</securitykey>
  <clientip>010.228.057.102</clientip>
  <giftaid>YES</giftaid>
  <paymentsystem>MC</paymentsystem>
  <paymentsystemdetails>Credit Card - Joint Stock Commercial Bank
"S,RU</paymentsystemdetails>
  <startdate/>
  <expirydate>0110</expirydate>
  <last4digits>0001</last4digits>
  <authprocessor>HSBC</authprocessor>
  <merchantnumber>84576401</merchantnumber>
  <accounttype>M</accounttype>
  <authattempt>1</authattempt>
```

Example request continued overleaf..

```
<vpsauthcode/>
<bankauthcode/>
<batchid/>
<billingfirstnames>test</billingfirstnames>
<billingsurname>test</billingsurname>
<billingaddress>test</billingaddress>
<billingaddress2/>
<billingcity>test</billingcity>
<billingstate/>
<billingpostcode>test</billingpostcode>
<billingcountry>GB</billingcountry>
<billingphone/>
<deliveryfirstnames>test</deliveryfirstnames>
<deliverysurname>test</deliverysurname>
<deliveryaddress>test</deliveryaddress>
<deliveryaddress2/>
<deliverycity>test</deliverycity>
<deliverystate/>
<deliverypostcode>test</deliverypostcode>
<deliverycountry>GB</deliverycountry>
<deliveryphone/>
<cardholder>test</cardholder>
<cardfirstnames>test</cardfirstnames>
<cardsurname>test</cardsurname>
<cardaddress>test</cardaddress>
<cardaddress2/>
<cardcity>test</cardcity>
<cardstate/>
<cardpostcode>test</cardpostcode>
<cardcountry>GB</cardcountry>
<customername/>
<customeremail/>
<contactnumber/>
<contactfax/>
<vendoremail/>
<systemused>T</systemused>
<vpsprotocol>2.23</vpsprotocol>
<callbackurl/>
<aborted/>
<released/>
<refunded>NO</refunded>
<repeated>NO</repeated>
<username>gbtest1</username>
<applyavscv2>0</applyavscv2>
<apply3dsecure>2</apply3dsecure>
<cv2result>NOTPROVIDED</cv2result>
<addressresult>NOTPROVIDED</addressresult>
<postcoderesult>NOTPROVIDED</postcoderesult>
<threedattempt>3</threedattempt>
<threedresult>NOTCHECKED</threedresult>
<eci/>
<cavv/>
<t3mscore/>
<t3maction>NORESULT</t3maction>
<t3mid/>
<basket/>
<emailmessage/>
<invalidstatus/>
<invalidstatusdetail/>
<invalidrequestsource/>
<locale>en</locale>
<xid/>
</vspaccess>
```

2.52 Reporting Command - getTransactionIPDetails

This command returns the IP Address and country of origin of the client browser used in the specified transaction. Only information about a transaction associated with the vendor can be returned. Transactions can be specified by either VendorTxCode or VPSTxID.

This is a subset of the getTransactionDetails request, to provide often requested functionality.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

See also: getTransactionSummary, getTransactionList, getBatchDetail

Request

Node/Element	Type	Description
<command>	String getTransactionIPDetails	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<vendortxcode> (optional) OR	String (40 characters)	The VendorTxCode of the transaction
<vpstxid> (optional)	String (40 characters)	The VPSTxID (transactionid) of the transaction

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<clientip> (optional)	String (15 characters)	The IP address of the customer associated with this transaction (where available)
<ipolocation> (optional)	String (100 characters)	The geolocation of the IP address (where available)

Example request on overleaf..

Sample XML Input

```
<vspaccess>
  <command>getTransactionIPDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vendortxcode>01Jan2010Transaction12345</vendortxcode>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Or

```
<vspaccess>
  <command>getTransactionIPDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vpstxid>00AA0A0A-0AA0-00A0-A000-A0000A0AA000</vpstxid>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <clientip>123.123.123.123</clientip>
  <iplocation>London</iplocation>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.53 Reporting Command - getTransactionList

This command returns a list of all transactions started between the specific dates for the given vendor. You can filter on transaction type, user name and success and failure. By default this command will return all successful transaction information, for all transaction types, between the specified dates.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

NB: The list is limited to 1 month periods to minimise impact on the database. Multiple queries would be needed to return results for longer periods.

NOTE: This command returns a maximum of 50 rows of information. The **<startrow>** and **<endrow>** fields show which rows have been returned, with **<totalrows>** showing the total amount of information available.

See also: [getTransactionSummary](#), [getTransactionDetail](#), [getBatchDetail](#)

Request

Node/Element	Type	Description
<command>	String getTransactionList	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startdate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time from which to begin searching for transactions
<enddate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time of the last transaction to be returned. Cannot be more than a month after the start date
<relatedtransactionid> (optional)	String (40 characters)	The vpstxid of a related transaction
<vendortxcode> (optional)	String (50 characters)	The vendortxcode of a transaction
<systemsused> (optional)	Node	
L<system> (optional)	String (1 character)	F = Form S = Server D = Direct T = Terminal
<accounttypes> (optional)	Node	
L<accounttype> (optional)	String (1 character)	E=Ecommerce M=Mail Order Telephone Order C=Continuous Authority
<username> (optional)	String (20 characters)	If specified, only the transactions entered by the specified user through Terminal will be returned.
<txtypes> (optional)	Node	Container for the transaction types being returned
L<txtype> (optional)	String (20 characters)	The transaction types supported. Contains PAYMENT, REFUND, DEFERRED, RELEASE, ABORT, REPEAT, REPEATDEFERRED, VOID, MANUAL, AUTHENTICATE, CANCEL or AUTHORISE

<result> (optional)	String (8 characters)	Should contain SUCCESS or FAILURE or INVALID to indicate if successful or failed or invalid transactions are being requested.
<paymentsystems> (optional)	node	
<paymentsystem> (optional)	String (7 characters)	Short name of card type
<amount> (optional)	Numeric (monetary value)	The value of the transaction
<currency> (optional)	String (3 characters)	
<last4digits> (optional)	Numeric (4 characters)	Last 4 digits of the card used for the transaction
<vpsauthcode> (optional)	Numeric (big integer value)	The unique Sage Pay auth code for a successful transaction only
<repeated> (optional)	String (3 characters)	YES for only repeated transactions, NO for those not yet repeated
<released> (optional)	String (3 characters)	YES for only released transactions, NO for those not yet released
<searchphrase> (optional)	String (100 characters)	The phrase to be search on – will search on customer names and address fields
<includeaddresses> (optional)	String (3 characters)	Either YES or NO
<invalids> (optional)	Numeric (1 character)	Either 0 = no invalids, 1 = include invalids, 2 = invalids only
<sorttype> (optional)	String (14 Characters)	Either ByDate or ByVendorTxCode
<sortorder> (optional)	String (4 Characters)	Either ASC or DESC
<startrow> (optional)	Numeric	
<endrow> (optional)	Numeric	

Response (continued overleaf)

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<transactions>	Node	A container for the list of transactions returned
<startrow>	Numeric (integer value)	The start row of the selected record set
<endrow>	Numeric (integer value)	The end row of the selected record set
<totalrows>	Numeric (integer value)	The total rows found based on the criteria provided
<transaction>	Node	A container for the individual transaction
<rownumber>	Numeric (integer value)	The current row number of the returned results
<vpstxid>	String (40 characters)	The unique VPSTxid (transaction id) of this transaction

L<vendortxcode>	String (40 characters)	The unique VendorTxCode of this transaction
L<Result>	String	Result of the transaction. Either SUCCESS or FAILURE
L<transactiontype>	String (15 characters)	The type of transaction. Values are: Authenticate, Authorise, Deferred, Manual, Payment, PreAuth, Refund, Repeat and RepeatDeferred
L<systemused>	String (1 character)	Contains F for Form, S for Server, D for Direct or T for Terminal.
L<accounttype>	String (1 character)	Contains E for eCommerce, M for Mail Order/Telephone Order or C for Continuous Authority.
L<amount>	Numeric (monetary value)	The value of the transaction
L<currency>	String (3 characters)	The ISO 4217 currency code of this transaction
L<started>	Datetime (dd/mm/yyyy hh:mm:ss)	The date/time that the transaction was registered with Sage Pay
L<vpsauthcode> (optional)	Numeric (big integer value)	The unique Sage Pay auth code for a successful transaction only
L<bankauthcode> (optional)	String (20 characters)	The authorisation code from the bank. For PayPal transactions this is the PayPal transaction ID
L<cv2result> (optional)	String (12 characters)	The results of the CV2 fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<addressresult> (optional)	String (12 characters)	The results of the AVS address fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<postcoderesult> (optional)	String (12 characters)	The results of the AVS postcode fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<threedresult> (optional)	String (12 characters)	The results of the 3D-Secure fraud checks. Either OK , NOTCHECKED , ATTEMPTONLY , NOTAVAILABLE , NOTAUTHED , INCOMPLETE or ERROR
L<cardholder> (optional)	String (100 characters)	If available, the name on the credit/debit card.
L<batchid> (optional)	Numeric (integer value)	If the transaction has been settled with the bank, this is the ID of that batch
L<aborted> (optional)	Datetime	The date/time this transaction was aborted, if applicable
L<released> (optional)	Datetime	The date/time this transaction was released, if applicable
L<refunded> (optional)	String (8 characters)	Either YES if fully refunded, NO if not refunded or PARTIAL if partly refunded
L<repeated> (optional)	String (3 characters)	Either YES if at least one REPEAT has occurred against this transaction, NO if none have
L<username> (optional)	String (20 characters)	If available, the name of the user account in MySagePay who processed this transaction.
L<t3mscore> (optional)	Numeric	A value in the range -100 to +100 as a fraud score from the third man, when available.
L<t3maction> (optional)	String (10 characters)	Either OK , HOLD or REJECT as a recommended course of action based on the fraud score.

L<t3mid> (optional)	String (20 characters)	The ID of this transaction in the 3 rd Man systems.
L<paymentsystem> (optional)	String (20 characters)	The card type used for the transaction
L<billingname> (optional)	String (100 characters)	If available, the name used for the billing address.
L<billingaddress> (optional)	String (500 characters)	If available, the address and postcode used for the billing address.
L<deliveryname> (optional)	String (100 characters)	If available, the name used for the delivery address.
L<deliveryaddress> (optional)	String (500 characters)	If available, the address and postcode used for the delivery address.
L<cardname> (optional)	String (100 characters)	If available, the name on the credit/debit card.
L<cardaddress> (optional)	String (500 characters)	If available, the address of the credit/debit card holder.
L<statusdetail> (optional)	String (300 characters)	the status detail of the transaction
L<authprocessor> (optional)	String (50 characters)	The name of the acquirer to whom this transaction's batch was sent

Sample XML Input

```

<vspaccess>
  <command>getTransactionList</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <startdate>01/01/2010 00:00:00</startdate>
  <enddate>02/01/2010 00:00:00</enddate>
  <relatedtransactionid></relatedtransactionid>
  <vendortxcode>transaction123</vendortxcode>
  <systemsused>
    <system>F</system>
    <system>S</system>
  </systemsused>
  <accounttypes>
    <accounttype>E</accounttype>
  </accounttypes>
  <username>Barry</username>
  <txtypes>
    <txtype>PAYMENT</txtype>
    <txtype>REFUND</txtype>
  </txtypes>
  <result>success</result>
  <paymentsystems>
    <paymentsystem>VISA</paymentsystem>
    <paymentsystem>MAESTRO</paymentsystem>
  </paymentsystems>
  <amount>10.00</amount>
  <currency>GBP</currency>
  <last4digits>1234</last4digits>
  <vpsauthcode>123456789</vpsauthcode>
  <repeated>YES</repeated>
  <released>YES</released>
  <searchphrase>John Smith</searchphrase>
  <includeaddresses>NO</includeaddresses>
  <sorttype>ByDate</sorttype>
  <sortorder>Desc</sortorder>
  <startrow>1</startrow>
  <endrow>50</endrow>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>

```

Example request continued overleaf..

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <transactions>
    <transaction>
      <rownumber>1</rownumber>
      <vpstxid>69042919-9710-06D1-0ED8-0AC146F9661F</vpstxid>
      <vendortxcode>REProtX.200909023502-961</vendortxcode>
      <result>SUCCESS</result>
      <transactiontype>Repeat</transactiontype>
      <location>MAIN</location>
      <systemused>S</systemused>
      <accounttype>C</accounttype>
      <amount>123.00</amount>
      <currency>GBP</currency>
      <started>09/09/2010 14:35:29.127</started>
      <vpsauthcode>552</vpsauthcode>
      <bankauthcode>999778</bankauthcode>
      <cv2result>NOTPROVIDED</cv2result>
      <addressresult>NOTPROVIDED</addressresult>
      <postcoderesult>NOTPROVIDED</postcoderesult>
      <threedresult>NOTCHECKED</threedresult>
      <cardholder>blah blah</cardholder>
      <batchid/>
      <aborted/>
      <released/>
      <refunded>NO</refunded>
      <repeated>NO</repeated>
      <username>rob</username>
      <t3mscore/>
      <t3maction>NORESULT</t3maction>
      <t3mid/>
      <paymentsystem>MasterCard</paymentsystem>
      <last4digits>0001</last4digits>
      <statusdetail/>
      <authprocessor>HSBC</authprocessor>
    </transaction>
    <startrow>1</startrow>
    <endrow>50</endrow>
  <totalrows>1</totalrows>
</transactions>
<timestamp>20/10/2010 17:42:56</timestamp>
</vspaccess>
```

2.54 Reporting Command - getTransactionSummary

This command returns a summary report of transactions started between the specific dates for the given vendor (equivalent to the transaction summary reports in MySagePay). You can filter on transaction type, user and success and failure. By default this command will return a summary of all successful transactions between the specified dates.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

NB: The list is limited to 1 month periods to minimise impact on the database. Multiple queries would be needed to return results for longer periods.

See also: getTransactionList, getTransactionDetail, getBatchDetail

Request

Node/Element	Type	Description
<command>	String getTransactionSummary	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startdate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time from which to begin searching for transactions
<enddate>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time of the last transaction to be returned. Cannot be more than a month after the start date
<result>	String (8 characters)	Should contain SUCCESS or FAILURE to indicate if successful or failed transactions are being requested.
<username> (optional)	String (20 characters)	If specified, only the transactions entered by the specified user through Terminal will be returned.
<txtypes> (optional)	Node	Container for the transaction types being returned
<txtype> (optional)	String (20 characters)	The transaction types supported. Contains PAYMENT, REFUND, DEFERRED, RELEASE, ABORT, REPEAT, REPEATDEFERRED, VOID, MANUAL, AUTHENTICATE, CANCEL or AUTHORISE

Response overleaf

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<users>	Node	A container for the list of summary records by user
L<user>	Node	A container for the individual user records
L<username>	String (20 characters)	The user who took this group of transactions
L<txtypes>	Node	Container for the transaction types being returned
L<txtype>	Node	Container for the transaction type being returned
L<transactiontype>	String (20 characters)	The type of transaction. Contains PAYMENT, REFUND, DEFERRED, RELEASE, ABORT, REPEAT, REPEATDEFERRED, VOID, MANUAL, AUTHENTICATE, CANCEL or AUTHORISE
L<currencies>	Node	A container for the currency list
L<currency>	String (3 characters)	Contains the ISO 4217 3-letter currency code of a currency for the transaction totals.
L<totalnumber>	Numeric (integer)	The number of transactions for this user, txtype and currency
L<totalvalue>	Numeric (monetary value)	The value of the transactions for this user, txtype and currency

Sample XML Input

```

<vspaccess>
  <command>getTransactionSummary</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <startdate>01/01/2010 00:00:00</startdate>
  <enddate>02/01/2010 00:00:00</enddate>
  <result>SUCCESS</result>
  <username>johnsmith</username>
  <txtypes>
    <txtype>PAYMENT</txtype>
  </txtypes>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>

```

Example request continued overleaf..

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <users>
    <user>
      <username>johnsmith</username>
      <txtypes>
        <txtype>
          <transactiontype>PAYMENT</transactiontype>
          <currencies>
            <currency>GBP</currency>
            <totalnumber>1</totalnumber>
            <totalvalue>10.00</totalvalue>
          </currencies>
        </txtype>
        <txtype>
          <transactiontype>REFUND</transactiontype>
          <currencies>
            <currency>GBP</currency>
            <totalnumber>1</totalnumber>
            <totalvalue>10.00</totalvalue>
          </currencies>
        </txtype>
      </txtypes>
    </user>
  </users>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.55 Reporting Command - getTransactionTypes

This command returns a list of all transaction types supported by the vendor account.



This command doesn't require the user to have administrator rights.

NB: There is no equivalent setTransactionTypes function. Additional transaction types must be requested from the Sage Pay support teams by e-mailing support@Sage Pay.com, including your Vendor Name and the transaction types you wish to have added.

See also: getURLs

Request

Node/Element	Type	Description
<command>	String getTransactionTypes	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<txtypes>	Node	Container for the transaction types being returned
L<txtype>	String (20 characters)	The transaction types supported. Contains PAYMENT, REFUND, DEFERRED, RELEASE, ABORT, REPEAT, REPEATDEFERRED, VOID, MANUAL, AUTHENTICATE, CANCEL or AUTHORISE

Sample Input XML:

```
<vspaccess>
  <command>getTransactionTypes</command>
  <vendor>sagepay</vendor >
  <user>sagepay</user>
</vspaccess>
```

Sample Output XML:

```
<vspaccess>
  <errorcode>0000</errorcode>
  <txtypes>
    <txtype>PAYMENT</txtype>
    <txtype>VOID</txtype>
    <txtype>REFUND</txtype>
  </txtypes>
</vspaccess>
```


2.56 Reporting Command - getRelatedTransactions

This command returns a list of all the related transactions of a specific transaction, and which started after a specific date for the given vendor.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

NOTE: This command returns a maximum of 50 rows of information. The `<startrow>` and `<endrow>` fields show which rows have been returned, with `<totalrows>` showing the total amount of information available.

Request

Node/Element	Type	Description
<code><command></code>	String getTransactionList	The service being requested.
<code><vendor></code>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<code><user></code>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<code><vpstxid></code>	String (40 characters)	The VPSTxID (transactionid) of the transaction
<code><startdate></code>	Datetime (dd/mm/yyyy hh:mm:ss)	The date and time from which to begin searching for transactions
<code><username></code> (optional)	String (20 characters)	If specified, only the transactions entered by the specified user through Terminal will be returned.

Response (continued overleaf)

Node/Element	Type	Description
<code><errorcode></code>	4-digit numeric	0000 if successful, positive value if an error occurred
<code><error></code> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<code><transactions></code>	Node	A container for the list of transactions returned
<code>L<startrow></code>	Numeric (integer value)	The start row of the selected record set
<code>L<endrow></code>	Numeric (integer value)	The end row of the selected record set
<code>L<totalrows></code>	Numeric (integer value)	The total rows found based on the criteria provided
<code>L<transaction></code>	Node	A container for the individual transaction
<code>L<rownumber></code>	Numeric (integer value)	The current row number of the returned results
<code>L<vpstxid></code>	String (40 characters)	The unique VPSTxId (transaction id) of this transaction
<code>L<vendortxcode></code>	String (40 characters)	The unique VendorTxCode of this transaction
<code>L<transactiontype></code>	String (15 characters)	The type of transaction. Values are: Authenticate, Authorise, Deferred, Manual, Payment, PreAuth, Refund, Repeat and RepeatDeferred
<code>L<systemused></code>	String (1 character)	Contains F for Form, S for Server, D for Direct or T for Terminal.

L<accounttype>	String (1 character)	Contains E for eCommerce, M for Mail Order/Telephone Order or C for Continuous Authority.
L<amount>	Numeric (monetary value)	The value of the transaction
L<currency>	String (3 characters)	The ISO 4217 currency code of this transaction
L<started>	Datetime (dd/mm/yyyy hh:mm:ss)	The date/time that the transaction was registered with Sage Pay
L<vpsauthcode> (optional)	Numeric (big integer value)	The unique Sage Pay auth code for a successful transaction only
L<bankauthcode> (optional)	String (20 characters)	The authorisation code from the bank. For PayPal transactions this is the PayPal transaction ID
L<cv2result> (optional)	String (12 characters)	The results of the CV2 fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<addressresult> (optional)	String (12 characters)	The results of the AVS address fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<postcoderesult> (optional)	String (12 characters)	The results of the AVS postcode fraud checks. Either MATCHED , NOTCHECKED , NOTPROVIDED , NOTMATCHED or PARTIAL
L<threeresult> (optional)	String (12 characters)	The results of the 3D-Secure fraud checks. Either OK , NOTCHECKED , ATTEMPTONLY , NOTAVAILABLE , NOTAUTHED , INCOMPLETE or ERROR
L<cardholder> (optional)	String (100 characters)	If available, the name on the credit/debit card.
L<batchid> (optional)	Numeric (integer value)	If the transaction has been settled with the bank, this is the ID of that batch
L<aborted> (optional)	Datetime	The date/time this transaction was aborted, if applicable
L<released> (optional)	Datetime	The date/time this transaction was released, if applicable
L<refunded> (optional)	String (8 characters)	Either YES if fully refunded, NO if not refunded or PARTIAL if partly refunded
L<repeated> (optional)	String (3 characters)	Either YES if at least one REPEAT has occurred against this transaction, NO if none have
L<username> (optional)	String (20 characters)	If available, the name of the user account in MySagePay who processed this transaction.
L<t3mscore> (optional)	Numeric	A value in the range -100 to +100 as a fraud score from the third man, when available.
L<t3maction> (optional)	String (10 characters)	Either OK , HOLD or REJECT as a recommended course of action based on the fraud score.
L<t3mid> (optional)	String (20 characters)	The ID of this transaction in the 3 rd Man systems.
L<billingname> (optional)	String (100 characters)	If available, the name used for the billing address.
L<billingaddress> (optional)	String (500 characters)	If available, the address and postcode used for the billing address.
L<deliveryname> (optional)	String (100 characters)	If available, the name used for the delivery address.

<code><deliveryaddress></code> (optional)	String (500 characters)	If available, the address and postcode used for the delivery address.
<code><cardname></code> (optional)	String (100 characters)	If available, the name on the credit/debit card.
<code><cardaddress></code> (optional)	String (500 characters)	If available, the address of the credit/debit card holder.
<code><statusdetail></code> (optional)	String (300 characters)	the status detail of the transaction

Sample XML Input

```
<vspaccess>
  <command>getRelatedTransactions</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <vpstxid>ABCEC4B5-A74F-40DB-89AA-6CEEC335D3FB</vpstxid>
  <startdate>01/01/2010 00:00:00</startdate>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <transactions>
    <transaction>
      <rownumber>1</rownumber>
      <vpstxid>ABCB1FC5-383A-E7C4-3307-000307BEB47D</vpstxid>
      <vendortxcode>My-d2cd51ea-d010-442a-b195-960190ec3fca</vendortxcode>
      <result>SUCCESS</result>
      <transactiontype>Authorise</transactiontype>
      <location>MAIN</location>
      <systemused>S</systemused>
      <accounttype>E</accounttype>
      <amount>36.50</amount>
      <currency>GBP</currency>
      <started>24/05/2010 19:37:46.567</started>
      <vpsauthcode>269992888</vpsauthcode>
      <bankauthcode>438888</bankauthcode>
      <cv2result>MATCHED</cv2result>
      <addressresult>MATCHED</addressresult>
      <postcoderesult>MATCHED</postcoderesult>
      <threedresult>ERROR</threedresult>
      <cardholder>Nicolas Nicholls</cardholder>
      <batchid>13833</batchid>
      <aborted/>
      <released/>
      <refunded>NO</refunded>
      <repeated>NO</repeated>
      <username />
      <t3mscore>-12</t3mscore>
      <t3maction>OK</t3maction>
      <t3mid>559527888</t3mid>
      <paymentsystem>MasterCard</paymentsystem>
      <last4digits>0001</last4digits>
      <statusdetail/>
      <authprocessor>NatWest Streamline</authprocessor>
    </transaction>
    <startrow>1</startrow>
    <endrow>50</endrow>
    <totalrows>1</totalrows>
  </transactions>
</vspaccess>
```

2.57 Reporting Command - getGiftAidReport

Returns a list of GiftAid transactions and the cardholders' names and addresses



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getGiftAid	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
<startdate>	String (23 characters)	The date at which to begin searching for Giftaid transactions.
<enddate>	String (23 characters)	The date at which to stop searching for Giftaid transactions

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<giftaidtransactions>	node	Contains the transaction nodes for transactions that had giftaid activated.
L<transaction>	Node	Details of a given transaction.
L<started>	Date (dd/mm/yyyy hh:mm:ss)	The date the transaction was processed.
L<vendortxcode>	String (50 characters)	The Vendor Chosen unique reference for this transaction.
L<amount>	Numerical	The value of the transaction.
L<currency>	String (3 characters)	Contains the ISO 4217 3-letter currency code of a supported currency. Under which the transaction was processed.
L<cardname>	String (100 characters)	The name of the card holder from the card on which the transaction was processed.
L<cardaddress>	String (500 characters)	The address of the card holder.

Sample Input XML:

```
<vspaccess>
  <command>getGiftAid</command>
  <vendor>sagepay</vendor>
  <user>charlie</user>
  <startdate>07-12-2009 00:00:00.000</startdate>
  <enddate>07-01-2010 00:00:00.000</enddate>
</vspaccess>
```

Example request continued overleaf..

Sample Response XML:

```
<vspaccess>
  <errorcode>0000</errorcode>
  <giftaidtransactions>
    <transaction>
      <started>08/12/2009 09:26:00</started>
      <vendortxcode>df1e7e11b832efc636a80e663313bf34f</vendortxcode>
      <amount>29.86</amount>
      <currency>GBP</currency>
      <cardname>C H Older</cardname>
      <cardaddress>41 Cholder Road, Lewisham, London SE13 1XF</cardaddress>
    </transaction>
    <transaction>
      <started>08/12/2009 09:26:00</started>
      <vendortxcode>rb894b7br83312832efc636a80e63421d</vendortxcode>
      <amount>10.25</amount>
      <currency>GBP</currency>
      <cardname>C U Stomer</cardname>
      <cardaddress>4 Board Road, Hoxton, London W1 BR8</cardaddress>
    </transaction>
  </giftaidtransactions>
</vspacceess>
```

2.58 Reporting Command - getAllAuthProcessors

This command gets all a list of all the auth processors (acquirers) which are associated with your account.

Request

Node/Element	Type	Description
<command>	String getAllAuthProcessors	The service being requested.
<vendor>	String (16 characters)	The Vendor Name
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<authprocessors>	Node	A container for the list of authprocessors returned
└<authprocessor>	String (50 characters)	The name of the acquirer to whom this batch was sent

Sample XML Input

```
<vspaccess>
  <command>getAllAuthProcessors</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <command>getAllAuthProcessors</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <authprocessors>
    <authprocessor>American Express</authprocessor>
    <authprocessor>Bank Of Scotland</authprocessor>
    <authprocessor>Barclays Merchant Services</authprocessor>
    <authprocessor>Diners Club</authprocessor>
    <authprocessor>Omnipay</authprocessor>
    <authprocessor>HSBC</authprocessor>
    <authprocessor>JCB</authprocessor>
    <authprocessor>Lloyds TSB Cardnet</authprocessor>
    <authprocessor>NatWest Streamline</authprocessor>
    <authprocessor>Alliance and Leicester</authprocessor>
  </authprocessors>
</vspaccess>
```

2.59 Reporting Command - getAllCountries

This command returns a list of all country codes.

Request

Node/Element	Type	Description
<command>	String getAllCountries	The service being requested.
<vendor>	String (16 characters)	The Vendor Name
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<countries>	Node	A container for the countries list.
L<country>	Node	A container for the country details.
L<countrycode>	String (2 characters)	Contains the ISO 2-letter country code of a country.
L<countryname>	String	The full name of the country.

Sample XML Input

```
<vspaccess>
  <command>getAllCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <command>getAllCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <countries>
    <country>
      <countrycode>AF</countrycode>
      <countryname>Afghanistan</countryname>
    </country>
    <country>
      <countrycode>AL</countrycode>
      <countryname>Albania</countryname>
    </country>
  </countries>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

2.60 Reporting Command - getBlockedCardRanges

This command returns all card BIN ranges currently in the blocked card range list on the vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getBlockedCardRanges	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<blockedcardranges>	node	Node containing blocked card range elements
L<blockedcardrange>	Numeric (9 characters)	The card BIN range to be blocked

Sample XML Input

```
<vspaccess>
  <command>getBlockedCardRanges</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <error></error>
  <blockedcardranges>
    <blockedcardrange>675967511</blockedcardrange>
  </blockedcardranges>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```


2.61 Reporting Command - getBlockedCountries

This command returns all card BIN ranges currently in the blocked card range list on the vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getBlockedCountries	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<blockedcountries>	node	Node containing blocked countries
L<countryrow>	node	Node containing country row elements
L<countrycode>	String (2 characters)	ISO3166 2 character country code
L<country>	String (100 characters)	The country name

Sample XML Input

```
<vspaccess>
  <command>addBlockedCountries</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <blockedcountries>
    <countryrow>
      <countrycode>ZW</countrycode>
      <country>Zimbabwe</country>
    </countryrow>
  </blockedcountries>
  <timestamp>20/10/2010 12:37:34</timestamp>
</vspaccess>
```

2.62 Reporting Command - getBlockedIPs

The getBlockedIPs command gets a list of the currently blocked IPs.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getBlockedIPs	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<ipaddresses>	Node	Node containing the ipaddress nodes
<ipaddress>	Node	Node containing the ipaddress and subnetmask values
<address>	String (15 characters)	The IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)
<mask>	String (15 characters)	The subnet mask for the IP address to be blocked in the format 123.123.123.123 – each number block must be 3 digits so zero padding may need to apply (i.e. 2.5.60.70 would be 002.005.060.070)

Sample XML Input

```
<vspaccess>
  <command>getBlockedIPs</command>
  <vendor>sagepay</vendor>
  <user>sagepay</user>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <ipaddresses>
    <ipaddress>
      <address>123.123.123.123</address>
      <mask>255.255.255.255</mask>
    </ipaddress>
  </ipaddresses>
</vspaccess>
```

2.63 Reporting Command - getBlockedIssuingCountries

Returns a list of the card issuing countries which will be prevented from processing transactions against this vendor account.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String	The service being requested.
<vendor>	String (16 characters)	The vendor account from which to retrieve the block issuing countries list.
<user>	String (20 characters)	Must be an admin user under the vendor account

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<blockedissuingcountries>	Node	The containing node of blocked issuing countries.
L<issuingcountry>	Node	The containing node of a single issuing country.
L<countrycode>	String (2 Characters)	ISO-2 character country code.
L<country>	String (100 characters)	The country name

Sample XML Input

```
<vspaccess>
  <command>getBlockedIssuingCountries</command>
  <vendor>egoo</vendor>
  <user>egoo</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <blockedissuingcountries>
    <issuingcountry>
      <countrycode>AT</countrycode>
      <country>Austria</country>
    </issuingcountry>
    <issuingcountry>
      <countrycode>BB</countrycode>
      <country>Barbados</country>
    </issuingcountry>
  </blockedissuingcountries>
  <timestamp>20/10/2010 12:49:49</timestamp>
</vspaccess>
```

2.64 Reporting Command - getTokenCount

This command returns the number of tokens the vendor currently has.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getTokenCount	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command. Must Have ADMIN rights on the account.

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<totalnumber>	Numeric	The total number of tokens held.

Sample XML Input

```
<vspaccess>
  <command>getTokenCount</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <totalnumber>255</totalnumber>
  <timestamp>25/03/09 15:16:50 GMT</timestamp>
</vspaccess>
```

2.65 Reporting Command - getTokenDetails

This command returns the details associated to a particular token.



The user name provided must have administrative privileges to run this command.

Request

Node/Element	Type	Description
<command>	String getTokenDetails	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command. Must Have ADMIN rights on the account.
<token>	String (40 characters)	The token you are requesting the details for

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<token>	String (40 characters)	The token details have been requested for
<cardholder>	String (50 characters)	The name of the cardholder
<last4digits>	String (4 characters)	Last 4 digits of the card associated with token
<paymentsystem>	String (10 characters)	The payment system associated with token. Contains either AMEX , DC (Diner's Club), DELTA (Visa Debit), JCB , MAESTRO , MC (MasterCard), PAYPAL , SOLO , SWITCH (UK Maestro), UKE (Visa Electron) or VISA (Visa Credit)
<corporatecard>	String (3 characters)	Will contain either YES or NO to indicate whether or not the card is a corporate card
<iscredit>	String (3 characters)	Will contain either YES or NO to indicate whether or not the card is a credit card
<expirydate>	String (4 characters)	The expiry date of card associated with token
<uses>	Numeric	No of attempts association with this token

Example request on overleaf..

Sample XML Input

```
<vspaccess>
  <command>getTokenDetails</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <token>4EBF8F56-82C4-4970-B584-461A44791ED3</token>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <paymentsystem>SOLO</paymentsystem>
  <last4digits>0005</last4digits>
  <cardholder>CHOlder</cardholder>
  <expirydate />
  <corporatecard>NO</corporatecard>
  <iscredit>NO</iscredit>
  <uses>2</uses>
  <timestamp>20/10/2010 17:19:52</timestamp>
</vspaccess>
```

2.66 Reporting Command - getT3MDetail

This command returns the third man fraud screening details for a particular transaction.



This command doesn't require the user to have administrator rights but only those transactions that can be viewed by the user account will be returned.

Request

Node/Element	Type	Description
<command>	String getT3MDetail	The service being requested.
<vendor>	String (16 characters)	The Vendor name of the Sage Pay account being accessed.
<user>	String (20 characters)	The User name, under the above referenced Vendor account, used to carry out this command.
< t3mtxid>	String (40 characters)	The unique transaction id of this transaction

Response

Node/Element	Type	Description
<errorcode>	4-digit numeric	0000 if successful, positive value if an error occurred
<error> (optional)	String (255 characters)	Human readable error message if errorcode is not 0000.
<t3mresults>	Node	Container for the list of third man rules.
└<rule>	Node	Container for a particular third man rule.
└<description>	Text	Text describing the rule.
└<score>	Numeric	The score for the associated rule.

Sample XML Input

```
<vspaccess>
  <command>getT3MDetail</command>
  <vendor>onlinecheese</vendor>
  <user>barry</user>
  <t3mtxid>559575366</t3mtxid>
  <signature>799B11DFF4275AEE76531AEC625FADE0</signature>
</vspaccess>
```

Example response on overleaf..

Sample XML Output

```
<vspaccess>
  <errorcode>0000</errorcode>
  <t3mresults>
    <rule>
      <description>Phone no is a UK landline</description>
      <score>-3</score>
    </rule>
    <rule>
      <description>CV2 Pass</description>
      <score>-10</score>
    </rule>
    <rule>
      <description>AVS Address pass</description>
      <score>-6</score>
    </rule>
    <rule>
      <description>AVS Postcode pass</description>
      <score>-6</score>
    </rule>
    <rule>
      <description>Internet transaction</description>
      <score>0</score>
    </rule>
    <rule>
      <description>AVS pass, bill = del, low val</description>
      <score>-5</score>
    </rule>
    <rule>
      <description>AVS pass, bill = del, low val</description>
      <score>-5</score>
    </rule>
    <rule>
      <description>AVS pass, bill = del, low val</description>
      <score>-5</score>
    </rule>
    <rule>
      <description/>
      <score/>
    </rule>
  </t3mresults>
  <timestamp>07/10/2010 10:03:39</timestamp>
</vspaccess>
```


3 Using Special Characters

In order to use the '&' character in a tag value the tag must be passed with CDATA. For example `<password><![CDATA[&egpassword1]]></password>`

This is required because otherwise the '&' character is interpreted as an XML value meaning that error 0001 will be displayed stating invalid XML was passed.

4 Full <errorcode> list

The table below lists all <errorcode> values that can be returned from the system, and the meaning of the code (attached in the <error> node)

Important: Please note that the wording of the error messages may change at a future date. Therefore, please ensure you reference the error codes themselves, *not* the error messages, when using these commands.

Code	Error Message
------	---------------

0000=	
0001=	The message received was not valid XML.
0002=	Invalid command. XML root node should be vspaccess.
0003=	No <command> element found.
0004=	<command> element either not found, or present more than once.
0005=	<vendor> element either not found, or present more than once.
0006=	<user> element either not found, or present more than once.
0007=	<signature> element either not found, or present more than once.
0008=	The specified <vendor> and <user> combination is not valid, or the user may be locked out.
0009=	Error (retrieving/updating or inserting) data. Please e-mail support@sagepay.com with this error.
0010=	Cannot validate the <signature> value.
0011=	You cannot execute the specified command without administrative privileges.
0012=	This command is restricted to Sage Pay only. Contact Sage Pay for more information.
0013=	The command passed in the <command> parameter was not recognised.
0014=	The createUser command requires a <newuser> element.
0015=	The <newuser> element requires a <username> value.
0016=	The <newuser> element requires a <password> value.
0017=	The <newuser> element requires a <accessrights> element.
0018=	The <accessrights> element requires at least one <access> value.
0019=	Error parsing the request XML. Check the format of your request message.
0020=	One of the <access> values you have supplied is not recognised.
0021=	Cannot create the new user. Check that the user does not already exist.
0022=	A valid <username> value is required.
0023=	Cannot (unlock,logout,lockout) the specified user.
0024=	Cannot change the specified user's password. Check that the user exists.
0025=	Cannot change user rights. Check that the user exists.
0026=	<status> element either not found, or present more than once.
0027=	The <status> value must be either ON or OFF.
0028=	Cannot set (AVSCV2/3DSecure) screening to (OFF/ON).
0029=	A valid <system> value is required.
0030=	You must include at least one <txtype> entry in a <txtypes> node.
0031=	The <system> value must be Form, Server or Direct.
0032=	You must include at least one <currency> entry in a <currencies> node.
0033=	You must include at least one <paymentsystem> entry in an <paymentsystems> node.
0034=	<accounttype> values must be E, C or M.
0035=	A valid <startdate> value is required.
0036=	A valid <enddate> value is required.
0037=	Cannot resolve either the start or end dates supplied.
0038=	The start date you have supplied is after the end date.
0039=	The end date is more than one month after the start date. You can only return reports at monthly intervals.
0041=	The <result> value should be either SUCCESS or FAILURE.
0042=	You must supply one of the following nodes with this command, either <vendortxcode> OR <vpstxid>.
0043=	Unable to find the transaction for the <vendortxcode> or <vpstxid> supplied.

0044=The user does not have permission to view this transaction.
0045=You cannot set the administrator password using vspaccess. Please contact support@sagepay.com if you need this reset.
0046=The <signature> does not match. Check you are using the correct password for this user.
0047=Cannot return the transaction types. Contact support@sagepay.com for more information.
0048=Cannot return the transaction list or summary. Contact support@sagepay.com for more information.
0049=Invalid <expirydate> supplied. It should be in MMY format.
0050=A valid <expirydate> value is required.
0051=You must provide the first 1-9 digits of the card number in the <cardbegins> field.
0052=Invalid <cardbegins> supplied. It should contain numbers only.
0053=A valid <batchid> value is required.
0054=A valid <authprocessor> value is required.
0055=A valid <batchid> values must be numeric.
0056=<username> must be alphanumeric and 20 characters or less.
0057=<password> must contain valid characters and be 32 characters or less.
0058=<username> element either not found, or present more than once.
0059=<startdate> element either not found, or present more than once.
0060=<enddate> element either not found, or present more than once.
0061=<batchid> element either not found, or present more than once.
0063=<enddate> cannot be before <startdate>, please check.
0064=<enddate> must be within 1 month of <startdate>, please check.
0065=<username> element either not found, or present more than once.
0066=<password> element either not found, or present more than once.
0067=A valid <securedisplayname> value is required.
0068=A valid <visamerchantid> value is required.
0069=A valid <mcmerchantid> value is required.
0070=<vendordetails> element either not found, or present more than once.
0071=<vendorprovidedname> element either not found, or present more than once.
0072=<homepageurl> element either not found, or present more than once.
0073=<supportemailaddress> element either not found, or present more than once.
0074=<sendhtmlemails> element either not found, or present more than once.
0075=<defaultterminalcurrency> element either not found, or present more than once.
0076=A valid <vendorprovidedname> value is required.
0077=A valid <homepageurl> value is required.
0078=A valid <supportemailaddress> value is required.
0079=A valid <sendhtmlemails> value is required.
0080=A valid <defaultterminalcurrency> value is required.
0081=<securedisplayname> element either not found, or present more than once.
0082=<visamerchantid> element either not found, or present more than once.
0083=<mcmerchantid> element either not found, or present more than once.
0084=Invalid <currency> supplied.
0085=Invalid <accounttype> supplied.
0086=A <system> element either not found, or present more than once.
0087=A valid <system> value is required.
0088=A valid <txtype> value is required, please check your request.
0089=<vspvendorname> element either not found, or present more than once.
0090=Invalid <vspvendorname> supplied.
0091=<txtypes> element either not found, or present more than once.
0093=Invalid <txtype> supplied.
0094=<vendors> element either not found, or present more than once.
0095=A <vspvendorname> element could not be found in the <vendors> element.
0096=Invalid vspvendorname value found in the vspvendorname elements.
0097=A valid <paypaluserid> value is required.
0098=Invalid <paypaluserid> supplied.
0099=Start Row and End Row range too large. The difference between the <startrow> and <endrow> value must be no greater than 50.
0200=A valid <vendorid> value is required.
0201=<blockedcountries> element either not found, or present more than once.
0202=You must provide at least one <countrycode> element within <blockedcountries> node.
0203=A valid <countrycode> value is required.
0204=<templatetype> element either not found, or present more than once.
0205=A valid <templatetype> value is required.
0206=<blockedissuingcountries> element either not found, or present more than once.
0207=<startvalue> element either not found, or present more than once.
0208=A valid <startvalue> value is required, this must be a numerical value between 0 and 100000.00.
0209=<endvalue> element either not found, or present more than once.
0210=A valid <endvalue> value is required, this must be a numerical value between 0 and 100000.00.
0211=<startvalue> value must be less than the <endvalue> value.
0212=<perform3dauth> element either not found, or present more than once.
0213=A valid <perform3dauth> value is required, this should be either YES or NO.
0214=<allownon3dcards> element either not found, or present more than once.

0215=A valid <allownon3dcards> value is required, this should be either YES or NO.
0216=<allownon3disuers> element either not found, or present more than once.
0217=A valid <allownon3disuers> value is required, this should be either YES or NO.
0218=<allow3dfailures> element either not found, or present more than once.
0219=A valid <allow3dfailures> value is required, this should be either YES or NO.
0220=<allowmpifailures> element either not found, or present more than once.
0221=A valid <allowmpifailures> value is required, this should be either YES or NO.
0222=<blockedcardranges> element either not found, or present more than once.
0223=At least 1 <blockedcardrange> element is required.
0224=A valid <blockedcardrange> value is required.
0225=<allownodatamatches> element either not found, or present more than once.
0226=A valid <allownodatamatches> value is required, this should be either YES or NO.
0227=<allowaddressmatchonly> element either not found, or present more than once.
0228=A valid <allowaddressmatchonly> value is required, this should be either YES or NO.
0229=<allowsecuritycodematchonly> element either not found, or present more than once.
0230=A valid <allowsecuritycodematchonly> value is required, this should be either YES or NO.
0231=<allowdatanotchecked> element either not found, or present more than once.
0232=A valid <allowdatanotchecked> value is required, this should be either YES or NO.
0233=<userpassword> element either not found, or present more than once.
0234=A valid <userpassword> value is required.
0235=More than 50,000 rows returned. Please refine the search.
0236=The <result> value should be either SUCCESS FAILURE or INVALID.
1001=Invalid <vspvendorname> supplied.
1100=The current vendor account is closed.
1101=The vendor account is suspended.
1103=The number of target users does not match targetvendor.
1104=More than one targetuser or targetvendor nodes supplied.
1105=<txtypes> element either not found, or present more than once.
1106=A valid <vendortxcode> value is required.
1107=A valid <vpstxid> value is required.
1108=A valid <access> value is required.
1109=<accessrights> element either not found, or present more than once.
1110=At least 1 <access> element is required.
1111=<merchantnumber> element either not found, or present more than once.
1112=Invalid <merchantnumber> supplied.
1113=<streamlinenumber> element either not found, or present more than once.
1114=Invalid <streamlinenumber> supplied.
1115=<streamlinenumber> element is only required for a Streamline merchant number.
1116=<accountnumber> element either not found, or present more than once.
1117=Invalid <accountnumber> supplied.
1118=<sortcode> element either not found, or present more than once.
1119=Invalid <sortcode> supplied.
1120=<accountcurrency> element either not found, or present more than once.
1121=Invalid <accountcurrency> supplied.
1122=<currencies> element either not found, or present more than once.
1123=You must include one <currencycode> entry in a <currency> node.
1124=Invalid <currencycode> supplied.
1125=You must include one <terminalids> node in a <currency> node.
1126=You must include at least one <terminalid> entry in a <terminalids> node.
1127=Invalid <terminalid> supplied.
1128=<accounttypes> element either not found, or present more than once.
1129=You must include at least one <accounttype> entry in a <accounttypes> node.
1130=<paymentsystems> element either not found, or present more than once.
1131=Invalid <paymentsystem> supplied.
1132=You must provide at least one <currency> element.
1133=<newuser> element either not found, or present more than once.
1134=<username> element either not found, or present more than once.
1135=<username> must be alphanumeric and with at least 1 character but no more than 20.
1136=<password> element either not found, or present more than once.
1137=A valid <password> value is required.
1138=<accessrights> element either not found, or present more than once.
1139=You must provide at least one <access> node within <accessrights> node.
1140=A valid <access> value is required.
1141=<vendordisplayname> element either not found, or present more than once.
1142=A valid <vendordisplayname> value is required.
1143=<companynumber> element either not found, or present more than once.
1144=A valid <companynumber> value is required.
1145=<vatnumber> element either not found, or present more than once.
1146=A valid <vatnumber> value is required.
1147=<encryptionpassword> element either not found, or present more than once.
1148=A valid <encryptionpassword> value is required.

1149=<reportingpassword> element either not found, or present more than once.
1150=A valid <reportingpassword> value is required.
1151=<actinicsecret> element either not found, or present more than once.
1152=A valid <actinicsecret> value is required.
1153=<currency> element either not found, or present more than once.
1154=A valid <currency> value is required.
1155=<visapassword> element either not found, or present more than once.
1156=<mcpassword> element either not found, or present more than once.
1157=<city> element either not found, or present more than once.
1158=A valid <city> value is required.
1159=<merchantcategorycode> element either not found, or present more than once.
1160=A valid <merchantcategorycode> value is required.
1161=<avscv2> element either not found, or present more than once.
1162=A valid <avscv2> value is required.
1163=<directrefund> element either not found, or present more than once.
1164=A valid <directrefund> value is required.
1165=<checkexpiryonrefund> element either not found, or present more than once.
1166=A valid <checkexpiryonrefund> value is required.
1167=<templatesuffix> element either not found, or present more than once.
1168=A valid <templatesuffix> value is required.
1169=<merchantaccounts> element either not found, or present more than once.
1170=You must include at least one <merchantaccount> within <merchantaccounts>.
1171=<authprocessor> element either not found, or present more than once.
1173=<streamlinenumber> element either not found, or present more than once.
1174=A valid <streamlinenumber> value is required.
1176=<ipaddresses> element either not found, or present more than once.
1177=You must include at least one <ipaddress> within <ipaddresses>.
1178=Target vendor could not be found.
1179=You must include both the <startrow> and <endrow> tags when selecting a range of batch rows.
1180=Invalid <batchtypeid> supplied.
1181=<batchtypeid> element either not found, or present more than once.
1184=You must include one <terminalids> node.
2100=A valid <vspvendorname> value is required.
2101=<blockedips> element either not found, or present more than once.
2102=You must provide at least one <ipaddress> node within <blockedips> node.
2103=<address> element either not found, or present more than once.
2104=A valid <address> value is required in the format 123.123.123.123. you may need to zero pad your ip address so that each number block contains three digits.
2105=<mask> element either not found, or present more than once.
2106=A valid <mask> value is required in the format 123.123.123.123. you may need to zero pad your subnet mask so that each number block contains three digits.
2107=A valid <terminalonly> value is required.
2108=Invalid <terminalonly> supplied.
2109=A valid <directrefunds> value is required.
2110=Invalid <directrefunds> supplied.
2111=A valid <giftaid> value is required.
2112=Invalid <giftaid> supplied.
2113=<validips> element either not found, or present more than once.
2114=You must provide at least one <ipaddress> node within <validips> node.
2115=<note> element either not found, or present more than once.
2116=A valid <note> value is required.
2117=<countrycode> element either not found, or present more than once.
2118=Invalid <securedisplayname> supplied.
2119=Invalid <visamerchantid> supplied.
2120=Invalid <mcmerchantid> supplied.
2121=Invalid <visapassword> supplied.
2122=Invalid <mcpassword> supplied.
2123=You must provide one <newvendor> tag.
2124=The <user> value was not supplied or in an invalid format.
2125=The <signature> value was not supplied.
2126=The <vendor> value was not supplied.
2127=Tag does not match protocol.
2128=The command cannot be run for the vspvendorname provided.
2129=<cardbegins> element either not found, or present more than once.
2130=<result> element either not found, or present more than once.
2131=A valid <startrow> value is required.
2132=A valid <endrow> value is required.
2135=<logentry> element either not found, or present more than once.
2136=Invalid <logentry> supplied (might be too long).
2138=A valid <currencycode> value is required.
2139=A valid <terminalid> value is required.

2140=A valid <accounttype> value is required.
2141=A valid <paymentsystem> value is required.
2142=<vendortxcode> or <vpstxid> element either not found, or present more than once.
2143=<vpstxid> element either not found, or present more than once.
2144=<vendortxcode> element either not found, or present more than once.
2145=A valid <systemused> value is required.
2146=<systemused> element either not found, or present more than once.
2147=<result> element either not found, or present more than once.
2148=A valid <amount> value is required.
2149=<amount> element either not found, or present more than once.
2150=You must include at least one <system> entry in an <systemused> node.
2151=<paymentsystems> element either not found, or present more than once.
2152=<currency> element either not found, or present more than once.
2153=A valid <last4digits> value is required.
2154=<last4digits> element either not found, or present more than once.
2155=A valid <vpsauthcode> value is required.
2156=<vpsauthcode> element either not found, or present more than once.
2157=A valid <repeated> value is required, this should be either YES or NO.
2158=<repeated> element either not found, or present more than once.
2159=A valid <released> value is required, this should be either YES or NO.
2160=<released> element either not found, or present more than once.
2161=A valid <searchphrase> value is required.
2162=<searchphrase> element either not found, or present more than once.
2163=A valid <includeaddresses> value is required, this should be either YES or NO.
2164=<includeaddresses> element either not found, or present more than once.
2165=A valid <invalids> value is required, this should be either 0, 1 or 2.
2166=<invalids> element either not found, or present more than once.
2167=A valid <sorttype> value is required, this should be either ByDate or ByVendorTxCode.
2168=<sorttype> element either not found, or present more than once.
2169=A valid <sortorder> value is required, this should be either ASC or DESC.
2170=<sortorder> element either not found, or present more than once.
2171=<homepage> element either not found, or present more than once.
2172=A valid <homepage> value is required.
2173=The user must have access to the specified <homepage>. Either provide the user with access to the chosen <homepage> or select another <homepage> which the user has access to.
2174=<accounttype> element either not found, or present more than once.
2175=<accounttypes> element either not found, or present more than once.
2176=A valid <relatedtransactionid> value is required.
2177=<relatedtransactionid> element either not found, or present more than once.
2178=<paymentsystem> element either not found, or present more than once.
2179=<newvendorid> element either not found, or present more than once.
2180=You cannot delete your own user.
2181=This is a restricted command. Contact Sage Pay for more information.
2182=Problem contacting the T3M Servers.
2183=<t3mtxid> element either not found, or present more than once.
2184=A valid <t3mtxid> value is required.
2185=PayPal couldn't verify the <paypaluserid>.
2186=<transactionid> element either not found, or present more than once.
2187=A valid <transactionid> value is required.
2188=<mask> is too restrictive. Only C class subnet masks are allowed for blocking.
2189=Could not add 3d secure details to this <merchantnumber>. Check that it is not an AMEX or PayPal one.
2190=Could not create new vendor. Check that it does not already exist.
2191=Could not get total authorisations for this transaction.
2192=Could not get total refunds for this transaction.
2193=3D Secure rule not added. Might overlap with an existing rule.
2194=AVSCV2 rule not added. Might overlap with an existing rule.
2195=Could not get currency limits for this currency. Currency might not exist.
2196=An admin account cannot run this command on itself.

8888=Unable to complete request. Please e-mail support@sagepay.com with this error.

8889=Unable to complete request. Please e-mail support@sagepay.com with this error.