Introduction

The Interspire Email Marketer XML API is a remotely accessible service API to allow Interspire Email Marketer users to run many of the Interspire Email Marketer API functions using XML requests.
The Interspire Email Marketer XML API makes it possible to programmatically update and use your system without needing to physically access it. As XML is a general purpose markup language you can use it to communicate between your Java based applications to Interspire Email Marketer's PHP application base.
For example you could set your system up to automatically update contact lists, create and send email campaigns, gather statistics and many other functions.

Requirements

To make use of the Interspire Email Marketer XML API you will need to be running PHP 5.1.2 or higher. This document is designed to explain the purpose of these functions and to provide examples of their use.

Submitting a Request

An XML POST request with the details for the license to be generated should be sent to the 'XML Path' that you can find in the 'User Accounts -> Edit User' section of Interspire Email Marketer under the 'User Permissions' tab. Make sure that you have 'Enable the XML API' checked and saved. The XML Path will look similar to the following:
http://www.yourdomain.com/IEM/xml.php
The 'username' and 'usertoken' mentioned in the following examples can be found in this same section under the title of 'XML Username' and 'XML Token' respectively.

Possible Requests

This section will describe the different functions that can be used by the Interspire Email Marketer XML API.

Add Subscriber to a List

The XML document structure for 'Adding a subscriber and associated custom details' request is as follows:/
- xmlrequest (Required)
  - username – The user name used to login to the Interspire Email Marketer. (Required)
  - usertoken – The unique token assigned to the user account used above. (Required)

- requesttype – The name of the API file in question. (Required)
- requestmethod – The name of the function being called. (Required)
- details (Required)
  - emailaddress – The email address of the contact being added. (Required)
  - mailinglistid – The list that the contact is located within. (Required)
  - confirmed – Sets the confirmation status of the subscriber to confirmed or not (yes or y or true or 1) (Not required, default to unconfirmed)
  - format – The format of the email campaigns that this contact prefers to receive (html or h or text or t) (defaults to text)
  - customfields
    - item
      - fieldid – The id of the custom field being added.
      - value – The value to be added to this custom field.

Successful Response

Upon submission of a valid 'add subscriber to list submission a contact will be added to the contact list and the contacts id number returned.

The format is as follows:

- response
  - status – The value of the status field will be "SUCCESS" for a successful response.
  - data – The contacts ID number.

Sample Request (XML)

The following code sample performs an insertion of the user 'email@domain.com' into the mailing list with ID '1' , status set to 'confirmed' , format set to 'html' and with a custom field set to 'John Smith'.

view plain   copy to clipboard   print   ?

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>d467e49b221137215ebdab1ea4e046746de7d0ea</usertoken>
04.    <requesttype>subscribers</requesttype>
05.    <requestmethod>AddSubscriberToList</requestmethod>
06.    <details>
07.      <emailaddress>email@domain.com</emailaddress>
08.      <mailinglist>1</mailinglist>
09.      <format>html</format>
10.      <confirmed>yes</confirmed>
11.      <customfields>
12.        <item>
13.          <fieldid>1</fieldid>
14.          <value>John Smith</value>
15.        </item>
16.      </customfields>
17.    </details>
18.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

view plain   copy to clipboard   print   ?

```
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>d467e49b221137215ebdab1ea4e046746de7d0ea</usertoken>
05.    <requesttype>subscribers</requesttype>
```

```
06.    <requestmethod>AddSubscriberToList</requestmethod>
07.    <details>
08.       <emailaddress>email@domain.com</emailaddress>
09.       <mailinglist>1</mailinglist>
10.       <format>html</format>
11.       <confirmed>yes</confirmed>
12.       <customfields>
13.          <item>
14.             <fieldid>1</fieldid>
15.             <value>John Smith</value>
16.          </item>
17.       </customfields>
18.    </details>
19. </xmlrequest>
20. ';
21.
22. $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
23. curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
24. curl_setopt($ch, CURLOPT_POST, 1);
25. curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
26. $result = @curl_exec($ch);
27. if($result === false) {
28.    echo "Error performing request";
29. }
30. else {
31.    $xml_doc = simplexml_load_string($result);
32.    echo 'Status is ', $xml_doc->status, '<br/>';
33.    if ($xml_doc->status == 'SUCCESS') {
34.       echo 'Data is ', $xml_doc->data, '<br/>';
35.    } else {
36.       echo 'Error is ', $xml_doc->errormessage, '<br/>';
37.    }
38. }
39.
40. ?>
```

Broken Response

This example will demonstrate an error message that is returned when a bad request is made.
Notice that the details tag in the example is not closed and no listid is sent for the function to use.
The XML document structure for this example is as follows:

- xmlrequest (Required)
  - username – The user name used to login to the Interspire Email Marketer. (Required)
  - usertoken – The unique token assigned to the user account used above. (Required)
  - requesttype – The name of the API file in question. (Required)
  - requestmethod – The name of the function being called. (Required)
  - details (Required)

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value the request will fail. A status message will be returned via XML as to why.

In this example you will be presented with the error message:

'The XML you provided is not valid. Please check your xml document and try again.'

The format is as follows:

- response
  - status – The value of the status field will be "ERROR".
  - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample will attempt to post a request to the Get Lists function.

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>lists</requesttype>
05.    <requestmethod>GetLists</requestmethod>
06.    </details>
07.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

```php
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
05.    <requesttype>lists</requesttype>
06.    <requestmethod>GetLists</requestmethod>
07.    </details>
08.  </xmlrequest>';
09.
10.  $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
11.  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
12.  curl_setopt($ch, CURLOPT_POST, 1);
13.  curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
14.  $result = @curl_exec($ch);
15.  if($result === false) {
16.    echo "Error performing request";
17.  }
18.  else {
19.    $xml_doc = simplexml_load_string($result);
20.    echo 'status is ', $xml_doc->status, '<br/>';
21.    if ($xml_doc->status == 'SUCCESS') {
22.      print_r($result);
23.      } else {
24.      echo 'Error is ', $xml_doc->errormessage, '<br/>';
25.    }
26.  }?>
```

Check Token Works

This example will check to make sure that the token that you are using is a valid token.
Notice that the details tag in the example is still included even though there are no details needed.

- xmlrequest (Required)
    - username – The user name used to login to the Interspire Email Marketer. (Required)
    - usertoken – The unique token assigned to the user account used above. (Required)
    - requesttype – The name of the API file in question. (Required)
    - requestmethod – The name of the function being called. (Required)
    - details (Required)

Successful Response

Upon submission of a valid 'check token' request the XML API will return true.
The format is as follows:

- response
    - status – The value of the status field will be "SUCCESS" for a successful response.
    - data – This will return '1' for a successful authorization.

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value, the Check Token will fail. A status message will be returned via XML as to why.
In this example you will be presented with the error message:
'Invalid details'
The format is as follows:

- response
    - status – The value of the status field will be "ERROR".
    - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample performs an authorization check on the usertoken to make sure that it is a valid token.

view plain   copy to clipboard   print   ?

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>authentication</requesttype>
05.    <requestmethod>xmlapitest</requestmethod>
06.    <details>
07.    </details>
08.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

view plain   copy to clipboard   print   ?

```
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
05.    <requesttype>authentication</requesttype>
06.    <requestmethod>xmlapitest</requestmethod>
07.    <details>
08.    </details>
09.  </xmlrequest>';
```

```
10.
11. $ch = curl_init('http://www.yourdomain.com/Interspire Email Marketer/xml.php');
12. curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
13. curl_setopt($ch, CURLOPT_POST, 1);
14. curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
15. $result = @curl_exec($ch);
16. if($result === false) {
17.    echo "Error performing request";
18. }
19. else {
20.    $xml_doc = simplexml_load_string($result);
21.    echo 'status is ', $xml_doc->status, '<br/>';
22.    if ($xml_doc->status == 'SUCCESS') {
23.       print_r($result);
24.       } else {
25.       echo 'Error is ', $xml_doc->errormessage, '<br/>';
26.    }
27. }
28. ?>
```

Delete Subscriber

This example will delete a contact from your contact list.
- xmlrequest (Required)
    - username – The user name used to login to the Interspire Email Marketer. (Required)
    - usertoken – The unique token assigned to the user account used above. (Required)
    - requesttype – The name of the API file in question. (Required)
    - requestmethod – The name of the function being called. (Required)
    - details (Required)
        - list – The ID of the Contact List that you are wishing to search (Required)
        - emailaddress – The email address of the contact that you are attempting to locate (Required)

Successful Response
Upon submission of a valid 'delete subscriber' request the XML API will return a status of successful.
The format is as follows:
- response
    - status – The value of the status field will be "SUCCESS" for a successful response.
    - data
        - item - the count of subscribers removed correctly.

Unsuccessful Response
Upon submission of an erroneous response due to a missing field, or an invalid value, the Delete Subscriber will fail. A status message will be returned via XML as to why.
The format is as follows:
- response
    - status – The value of the status field will be "ERROR".
    - errormessage – A textual message explaining why the request failed.

Sample Request (XML)
The following code sample will delete the email address listed from the contact list of this ID.

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>subscribers</requesttype>
05.    <requestmethod>DeleteSubscriber</requestmethod>
06.    <details>
07.      <emailaddress>email@domain.com</emailaddress>
08.      <list>1</list>
09.    </details>
10.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

```php
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
05.    <requesttype>subscribers</requesttype>
06.    <requestmethod>DeleteSubscriber</requestmethod>
07.    <details>
08.      <emailaddress>email@domain.com</emailaddress>
09.      <list>1</list>
10.    </details>
11.  </xmlrequest>';
12.
13.  $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
14.  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
15.  curl_setopt($ch, CURLOPT_POST, 1);
16.  curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
17.  $result = @curl_exec($ch);
18.  if($result === false) {
19.    echo "Error performing request";
20.  }
21.  else {
22.    $xml_doc = simplexml_load_string($result);
23.    echo 'status is ', $xml_doc->status, '<br/>';
24.    if ($xml_doc->status == 'SUCCESS') {
25.      print_r($result);
26.    } else {
27.      echo 'Error is ', $xml_doc->errormessage, '<br/>';
28.    }
29.  }
30.  ?>
```

Get Custom Field Data

The XML document structure for retrieving a contact lists custom field data is as follows:

- xmlrequest (Required)

- username – The user name used to login to the Interspire Email Marketer. (Required)
- usertoken – The unique token assigned to the user account used above. (Required)
- requesttype – The name of the API file in question. (Required)
- equestmethod – The name of the function being called. (Required)
- details (Required)
  - listids –The ID's of the contact list that the custom field data is being requested. (Required)

Successful Response

Upon submission of a valid 'Get Custom Field Data' submission the custom field data found will be returned in formatted XML.

The format is as follows:
- response
  - status – The value of the status field will be "SUCCESS" for a successful response.
  - data
  - item
    - fieldid – The custom fields ID number.
    - name – The name of the custom field.
    - fieldtype – the type of field (text, number etc.).
    - defaultvalue – If you set a default value it will appear here.
    - required – If this field is required to be filled in (1 or 0).
    - fieldsettings – Serialized version of the custom fields settings

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value the Get Custom Filed Data will fail. A status message will be returned via XML as to why.

The format is as follows:
- response
  - status – The value of the status field will be "ERROR".
  - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample will draw any information found on the custom fields for the list with the ID of '1'.

view plain   copy to clipboard   print   ?

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>lists</requesttype>
05.    <requestmethod>GetCustomFields</requestmethod>
06.    <details>
07.      <listids>
08.        1
09.      </listids>
10.    </details>
11.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

view plain   copy to clipboard   print   ?

```
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
```

```
05.      <requesttype>lists</requesttype>
06.      <requestmethod>GetCustomFields</requestmethod>
07.      <details>
08.        <listids>
09.          1
10.        </listids>
11.      </details>
12.  </xmlrequest>
13.  ';
14.
15.  $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
16.  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
17.  curl_setopt($ch, CURLOPT_POST, 1);
18.  curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
19.  $result = @curl_exec($ch);
20.  if($result === false) {
21.      echo "Error performing request";
22.  }
23.  else {
24.      $xml_doc = simplexml_load_string($result);
25.      echo 'status is ', $xml_doc->status, '<br/>';
26.      if ($xml_doc->status == 'SUCCESS') {
27.         print_r($result);
28.         } else {
29.         echo 'Error is ', $xml_doc->errormessage, '<br/>';
30.      }
31.  }
32.  ?>
```

Get Lists

This example will retrieve a list of all the Contact Lists that are located in Interspire Email Marketer and
return formatted XML with the data found.
Notice that the details tag in the example is still included even though there are no details needed.

- xmlrequest (Required)
  - username – The user name used to login to the Interspire Email Marketer. (Required)
  - usertoken – The unique token assigned to the user account used above. (Required)
  - requesttype – The name of the API file in question. (Required)
  - requestmethod – The name of the function being called. (Required)
  - details (Required)

Successful Response
Upon submission of a valid 'get lists' request the XML API will return any data found on the Interspire
Email Marketer Contact Lists will be returned in formatted XML.
The format is as follows:

- response
  - status – The value of the status field will be "SUCCESS" for a successful response.
  - data
    - item
      - listid – The id of the list.

- name – The name of the list.
- subscribecount – A count of how many subscribed contacts.
- unsubscribecount – A count of how many unsubscribed contacts.
- autorespondercount – A count of how many autoresponders are linked to the Contact List.

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value, the Get Lists will fail. A status message will be returned via XML as to why.

In this example you will be presented with the error message:

'Invalid details'

The format is as follows:

- response
    - status – The value of the status field will be "ERROR".
    - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample performs a request to return all the Contact Lists located within Interspire Email Marketer.

view plain   copy to clipboard   print   ?

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>user</requesttype>
05.    <requestmethod>GetLists</requestmethod>
06.    <details>
07.    </details>
08.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

view plain   copy to clipboard   print   ?

```
01.  <?php
02.  $xml = '<xmlrequest>
03.    <username>admin</username>
04.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
05.    <requesttype>user</requesttype>
06.    <requestmethod>GetLists</requestmethod>
07.    <details>
08.    </details>
09.  </xmlrequest>';
10.
11.  $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
12.  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
13.  curl_setopt($ch, CURLOPT_POST, 1);
14.  curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
15.  $result = @curl_exec($ch);
16.  if($result === false) {
17.    echo "Error performing request";
18.  }
19.  else {
20.    $xml_doc = simplexml_load_string($result);
21.    echo 'status is ', $xml_doc->status, '<br/>';
```

```
22.    if ($xml_doc->status == 'SUCCESS') {
23.        print_r($result);
24.        } else {
25.        echo 'Error is ', $xml_doc->errormessage, '<br/>';
26.    }
27. }
28. ?>
```

Get Subscribers

This example will retrieve a count of the amount of Contacts on the list in question as well as a list of all the Contacts that are located in that list and return formatted XML with the data found.

- xmlrequest (Required)
  - username – The user name used to login to the Interspire Email Marketer. (Required)
  - usertoken – The unique token assigned to the user account used above. (Required)
  - requesttype – The name of the API file in question. (Required)
  - requestmethod – The name of the function being called. (Required)
  - details (Required)
    - searchinfo
      - List – The ID of the Contact List that you are wishing to search (Required)
      - Email – The email address of the contact that you are attempting to locate (Required)

Successful Response

Upon submission of a valid 'get subscribers' request the XML API will return a count of the amount of Contacts on the list in question along with a list of those contact email addresses in formatted XML.
The format is as follows:

- response
  - status – The value of the status field will be "SUCCESS" for a successful response.
  - data
    - count – A count of the amount of subscribers.
    - subscriberlist – A list of the subscribers email addresses.

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value, the Get Subscribers will fail. A status message will be returned via XML as to why.
The format is as follows:

- response
  - status – The value of the status field will be "ERROR".
  - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample performs a check on the Contact List with ID '1' for all email addresses that contain the information '@yourdomain.com'.

```
01. <xmlrequest>
02.     <username>admin</username>
03.     <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.     <requesttype>subscribers</requesttype>
05.     <requestmethod>GetSubscribers</requestmethod>
06.     <details>
```

```
07.        <searchinfo>
08.          <List>1</List>
09.          <Email>@yourdomain.com</Email>
10.        </searchinfo>
11.      </details>
12.   </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

```php
01. <?php
02. $xml = '<xmlrequest>
03.      <username>admin</username>
04.      <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
05.      <requesttype>subscribers</requesttype>
06.      <requestmethod>GetSubscribers</requestmethod>
07.      <details>
08.        <searchinfo>
09.          <List>1</List>
10.          <Email>@yourdomain.com</Email>
11.        </searchinfo>
12.      </details>
13. </xmlrequest>';
14.
15. $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
16. curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
17. curl_setopt($ch, CURLOPT_POST, 1);
18. curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
19. $result = @curl_exec($ch);
20. if($result === false) {
21.    echo "Error performing request";
22. }
23. else {
24.    $xml_doc = simplexml_load_string($result);
25.    echo 'status is ', $xml_doc->status, '<br/>';
26.    if ($xml_doc->status == 'SUCCESS') {
27.       print_r($result);
28.       } else {
29.       echo 'Error is ', $xml_doc->errormessage, '<br/>';
30.    }
31. }
32. ?>
```

Is Contact on List

This example will check to see if a particular Contact is located on a particular Contact List.
- xmlrequest (Required)
    - username – The user name used to login to the Interspire Email Marketer. (Required)
    - usertoken – The unique token assigned to the user account used above. (Required)

- requesttype – The name of the API file in question. (Required)
- requestmethod – The name of the function being called. (Required)
- details (Required)
  - Email – The email address of the Contact you are searching for (Required)
  - List – The ID of the Contact List you are wanting to search (Required)

Successful Response

Upon submission of a valid 'is contact on list' request the XML API will return true if it locates the contact and will return nothing if not.
The format is as follows:
- response
  - status – The value of the status field will be "SUCCESS" for a successful response.
  - data – Will return '1' if contact is located.

Unsuccessful Response

Upon submission of an erroneous response due to a missing field, or an invalid value, the Is Contact on List will fail. A status message will be returned via XML as to why.
The format is as follows:
  response
  - status – The value of the status field will be "ERROR".
  - errormessage – A textual message explaining why the request failed.

Sample Request (XML)

The following code sample performs a check on a contact list to see if a particular contact is listed on it.

view plain   copy to clipboard   print   ?

```
01.  <xmlrequest>
02.    <username>admin</username>
03.    <usertoken>78701d1ba2588ea1991f44299b814129658c6947</usertoken>
04.    <requesttype>subscribers</requesttype>
05.    <requestmethod>IsSubscriberOnList</requestmethod>
06.    <details>
07.      <Email>email@yourdomain.com</Email>
08.      <List>1</List>
09.    </details>
10.  </xmlrequest>
```

Sample Request (PHP)

The following sample code is written in PHP and makes use of PHP's CURL functionality to insert the above XML into the application.

view plain   copy to clipboard   print   ?

```
01.  <?php
02.  $xml = ' <requesttype>subscribers</requesttype>
03.      <requestmethod>IsSubscriberOnList</requestmethod>
04.      <details>
05.        <Email>email@yourdomain.com</Email>
06.        <List>1</List>
07.      </details>
08.  </xmlrequest>';
09.
10.  $ch = curl_init('http://www.yourdomain.com/IEM/xml.php');
11.  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
12.  curl_setopt($ch, CURLOPT_POST, 1);
13.  curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);
14.  $result = @curl_exec($ch);
15.  if($result === false) {
```

```
16.      echo "Error performing request";
17.  }
18.  else {
19.      $xml_doc = simplexml_load_string($result);
20.      echo 'status is ', $xml_doc->status, '<br/>';
21.      if ($xml_doc->status == 'SUCCESS') {
22.          print_r($result);
23.          } else {
24.          echo 'Error is ', $xml_doc->errormessage, '<br/>';
25.      }
26.  }
27.  ?>
```

If you have any questions or problems regarding the Interspire Email Marketer API please leave a comment below or click here to submit a ticket through our client area.