## Order System

<table>
<thead>
<tr>
<th>Table of Contents</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• 1 About Order System
• 2 Tie-Ins*
• 3 Installing Order System
  • 3.1 Order System Overview
• 4 Creating an Order Form
  • 4.1 Basic
  • 4.2 Basic (Continued)
  • 4.3 Package Groups & Currencies
  • 4.4 Settings
    • 4.4.1 Basic Options
    • 4.4.2 Anti-Fraud
    • 4.4.3 Embed Code
    • 4.4.4 Marketing
• 5 Order Emails
  • 5.1 Order Received
  • 5.2 Order Received (Mobile)
• 6 Linking to Order Pages
  • 6.1 Linking to a Specific Order Form
  • 6.2 Linking to a Specific Product
  • 6.3 Linking to an Order Form to Include a Coupon
  • 6.4 Linking to an Order Form to Include the Currency
  • 6.5 Linking to an Order Form to Include a Specific Language
• 7 Activating the Affiliate System
• 8 Common Issues
About Order System

The Order System is responsible for all order forms and order handling in Blesta. As of version 4.11, the Order System also includes an Affiliate System, which is tightly integrated with orders.

Tie-Ins*

<table>
<thead>
<tr>
<th>Widgets</th>
<th>Nav Links</th>
<th>Client Cards</th>
<th>Automation Tasks</th>
<th>Email Templates</th>
<th>ACL Permissions</th>
<th>Search</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>

* Plugins are powerful and are not limited to the tie-ins listed above, these are just some of the most common

Installing Order System

1. Visit [Settings] > [Company] > [Plugins] > Available.
2. Click the “Install” button within the Order System plugin listing.
3. To enable the Order System widget, go to [Billing] > Overview, click the “Manage Widgets” link, and drag the Orders widget from the right to the left.

Order System Overview

Once installed, a new link will appear under [Packages] > Order Forms, where order forms can be managed. In the screen below, two order forms have already been created and they can be managed at this location.

Creating an Order Form

To create an order form, click the “Add Order Form” button in the upper right of the overview and fill out the options. Click “Add Form” to create the order form.

Basic

This section requests some basic information regarding the order form and its configuration including:

1. Status – This is the status of this order form, Active to enable the order form or Inactive to disable it.
2. Name – The name of this order form, this will appear in the heading on front facing pages for this order form.
3. Label – The label helps make up the URL to this order form, which is displayed next to the field, and must be unique.
4. Visibility – The visibility controls whether the order form appears on the order forms listing page and whether its packages may be ordered.
   a. Public – The order form is always visible and orderable for everyone
   b. Shared – The order form is not visible on the order form listing page, but may be orderable for everyone, such as via a direct link to the order form
   c. Client Only – The order form is not visible on the order form listing page and may only be orderable by clients that are currently logged in
5. Type – This is the type of order form, currently the following three are available: Domain and Other, General, and Client Registration. In most cases, the General type is preferred. This selection will determine what options are displayed for the Template.
6. Template – Select an order form template. The template controls the look and flow of the order form. Wizard boxes is the default template, and is most widely used.
Basic (Continued)

This section contains some additional options within the Basic section.

1. Default Client Group – This is the client group users who sign up through this order form will be assigned.
2. Allow Coupons – Whether or not coupons should be allowed on this order form. If coupons are not allowed, the coupon section on the order form will not be displayed.
3. Require Manual Review and Approval of All Orders – If this is checked, all orders will need to be approved by a staff member before being provisioned, regardless of any anti-fraud settings.
4. Force Secure Connection (HTTPS) – If checked, the order form will force https:// URLs.
5. Require Human Verification Challenge for All Signups – If checked, reCaptcha human verification will be used, depending on the Setting under the Settings tab.
6. Require Agreement to Terms of Service – If checked, users will be required to agree to the terms of service before checking out.
7. Terms of Service URL – Enter the URL to your terms of service page. This isn’t necessary if “Require Agreement to Terms of Services” isn’t checked.

Package Groups & Currencies

These sections determine what Package Groups (And hence Packages) are available through this order form as well as what currencies are accepted.

1. Assigned Groups & Available Groups – Move Package Groups from the right box (Available Groups) to the left box (Assigned Groups) to make them available on this order form. As of v4.3.0 Package Groups are displayed on the order form in the order shown here under ‘Assigned Groups’.
2. Currencies – Select which currencies to accept on this order form. If only one currency is selected, the currency selector will not appear on the order form.
Settings

The Order System supports various settings for human verification, fraud detection, embed codes, marketing emails, and allows the default order form to be selected. The default order form appears at ~/order/ and is linked to from the Portal page. Settings consist of several sections.

Basic Options

1. Default Order Form – Select which order form should be the default order form. It will be available at ~/order/ and linked to directly from the portal page. Alternatively, select "None - Show Listing" which will instead show a listing of all order forms at ~/order/.
2. Human Verification – Select from None or reCaptcha. None disables human verification, reCaptcha v2 is an option to prevent bots from signing up. Each require an account and additional unique settings which can be obtained from them. reCaptcha requires a Site Key and Shared Key.

Anti-Fraud

Type – The type of anti-fraud to use. The current options are None, Maxmind, and FraudLabsPro. None disables anti-fraud checks, Maxmind enables them through Maxmind, and FraudLabsPro enabled them through FraudLabsPro. An account with Maxmind is required to use Maxmind anti-fraud. An account with FraudLabsPro is required to use FraudLabsPro anti-fraud. The following options assume that Maxmind has been selected.

The following settings assume Maxmind was selected:

1. Type: Maxmind
2. Anti-Fraud Frequency – When to run anti-fraud checks. It is recommended to select the option to "Run fraud checks for all orders and customer signups".
3. Server – The server URL for Maxmind, this is typically: minfraud.maxmind.com
4. License Key – The license key obtained from Maxmind.
5. Minimum Score to Reject – The minimum score to reject an order. This should typically be set high, 80 is the default value. If Maxmind reports a value equal to or greater than this, the order will be outright rejected.
6. Minimum Score to Review – This should be lower than the minimum score to reject, default is 10. If the score of an order is equal to or greater than this but lower than the minimum score to reject, the order will be flagged for review. When an order is flagged for review it will remain as a Pending order until moved to either the Accepted or Fraud category by a staff user, at which time the client account will be automatically marked as active or fraud.

The following settings assume FraudLabsPro was selected:

1. Type: FraudLabsPro
2. Anti-Fraud Frequency – when to run anti-fraud checks, it's recommended to "Run fraud checks for all orders and customer signups".
3. API Key – The API key you obtained from FraudLabsPro
4. Minimum Score to Reject – The minimum score to reject an order. This should typically be set high, 80 is the default value. If FraudLabsPro reports a value equal to or greater than this, the order will be outright rejected.
5. **Minimum Score to Review** – This should be lower than the minimum score to reject, default is 10. If the score of an order is equal to or greater than this but lower than the minimum score to reject, the order will be flagged for review. When an order is flagged for review it will remain as a Pending order until moved to either the Accepted or Fraud category by a staff user, at which time the client account will be automatically marked as active or fraud.

6. **Automatically Reject/Review based on recommendation from FraudLabs Pro** – Yes is recommended, result will depend on what FraudLabs Pro suggests.

**Embed Code**

The embed code field allows any HTML or JavaScript to be directly injected, as written, into the footer of pages on the order form. Tag replacement is supported (see Customizing Emails), including a tag for the `order_page` so that you may set content only on specific pages.

The "order_page" tag contains the controller and action that references a specific page, which may include any one of the following:

<table>
<thead>
<tr>
<th>{{order_page}} value</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>signup/index</td>
<td>The new account registration page</td>
</tr>
<tr>
<td>main/index</td>
<td>The package selection page</td>
</tr>
<tr>
<td>main/packages</td>
<td>A specific package selection page</td>
</tr>
<tr>
<td>forms/index</td>
<td>The order form listing page</td>
</tr>
<tr>
<td>config/index</td>
<td>A specific package configuration page</td>
</tr>
<tr>
<td>config/preconfig</td>
<td>A specific package pre-configuration page</td>
</tr>
<tr>
<td>checkout/index</td>
<td>The payment page</td>
</tr>
<tr>
<td>checkout/complete</td>
<td>The completed order page</td>
</tr>
<tr>
<td>cart/index</td>
<td>The cart page</td>
</tr>
</tbody>
</table>

For example, the following embed code will display a message at the bottom of the registration page. You can use the `{% debug %}` tag to display all data available to aid in determining which variables you want to use, but this tag should not be used in production. Note that the data available will change depending on what stage in the order process you are.

```
{% if order_page == "signup/index" %}
  <h1>We are on the signup page</h1>
{% endif %}
```

**Marketing**

While the Order plugin does not send marketing emails, prospective clients may need to opt-in or opt-out of receiving marketing emails that you or the system may later send.

**Order Emails**

Order related email templates can be found under [Settings] > [Company] > [Emails] > Email Templates, in the section labeled "Plugin Templates".

**Order Received**

The order received email templates allow for the following tags:

<table>
<thead>
<tr>
<th>Tags</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>(order.id)</td>
<td>The system-level order ID</td>
<td></td>
</tr>
<tr>
<td>(order.number)</td>
<td>The order number</td>
<td>e.g. &quot;519131eb623b8&quot;</td>
</tr>
<tr>
<td>(order.client_id)</td>
<td>The system-level client ID of the client this order belongs to</td>
<td></td>
</tr>
<tr>
<td>(order.client_id_code)</td>
<td>The friendly client ID code</td>
<td></td>
</tr>
<tr>
<td>(order.client_first_name)</td>
<td>The first name of the client</td>
<td></td>
</tr>
<tr>
<td>(order.client_last_name)</td>
<td>The last name of the client</td>
<td></td>
</tr>
<tr>
<td>(order.client_company)</td>
<td>The client's company name</td>
<td></td>
</tr>
<tr>
<td>(order.client_address1)</td>
<td>The client's primary address</td>
<td></td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>order.client_email</td>
<td>The client's email address</td>
<td></td>
</tr>
<tr>
<td>invoice.id</td>
<td>The system-level invoice ID of the invoice created for this order</td>
<td></td>
</tr>
<tr>
<td>invoice.id_code</td>
<td>The friendly invoice ID code</td>
<td></td>
</tr>
<tr>
<td>invoice.date_due</td>
<td>The UTC datetime stamp of the invoice due date</td>
<td></td>
</tr>
<tr>
<td>invoice.date_billed</td>
<td>The UTC datetime stamp of the invoice bill date</td>
<td></td>
</tr>
<tr>
<td>invoice.total</td>
<td>The total amount due on the invoice</td>
<td></td>
</tr>
<tr>
<td>invoice.paid</td>
<td>The total amount paid</td>
<td></td>
</tr>
<tr>
<td>invoice.due</td>
<td>The remaining amount due</td>
<td></td>
</tr>
<tr>
<td>invoice.currency</td>
<td>The currency that the invoice is in</td>
<td></td>
</tr>
</tbody>
</table>

Due to the nature of tag objects containing several fields, many of which are likely irrelevant for use in email templates, but may be useful to you in certain circumstances, a dump of the tags are shown below.

### The (order) object

```
stdClass Object
{
    [id] => 52
    [order_number] => 519131eb623b8
    [order_form_id] => 1
    [invoice_id] => 19983
    [fraud_report] =>
    [status] => pending
    [date_added] => 2013-05-13 18:33:15
    [fraud_status] =>
    [order_form_label] => my_order_form
    [order_form_name] => My Order Form
    [client_id] => 3
    [total] => 8.0000
    [paid] => 0.0000
    [currency] => USD
    [date_closed] =>
    [invoice_id_code] => PD-15695
    [client_id_code] => PD-1501
    [client_first_name] => Mark
    [client_last_name] => Stevens
    [client_company] => Comp Co.
    [client_address1] => 123 Test St.
    [client_email] => user@domain.com
    [services] => Array
        {
            [0] => stdClass Object
                {
                    [order_id] => 52
                    [service_id] => 234
                }
        }
}
```

### The (invoice) object

```
stdClass Object
{
    [id] => 19983
    [id_format] => PD-(num)
    [id_value] => 15695
    [client_id] => 3
    [date_billed] => 2013-05-13 18:33:15
    [date_due] => 2013-05-13 18:33:15
    [date_closed] =>
    [date_autodebit] =>
```

```
The `{services}` array of objects

Array

```php

[0] => stdClass Object

  [id] => 234
  [parent_service_id] =>
  [package_group_id] => 1

```
[id_format] => PD-{num}
[id_value] => 229
[pricing_id] => 22
[module_row_id] => 6
[coupon_id] =>
[qty] => 1
[status] => in_review
[date_renews] => 2013-06-13 18:33:14
[date_last_renewed] =>
[date_suspended] =>
[date_canceled] =>
[id_code] => PD-229
[fields] => Array
 |
 | [0] => stdClass Object
 | |
 | | [key] => cpanel_confirm_password
 | | [value] => password
 | | [serialized] => 0
 | | [encrypted] => 1
 |
 | [1] => stdClass Object
 | |
 | | [key] => cpanel_domain
 | | [value] => domain.com
 | | [serialized] => 0
 | | [encrypted] => 0
 |
 | [2] => stdClass Object
 | |
 | | [key] => cpanel_password
 | | [value] => password
 | | [serialized] => 0
 | | [encrypted] => 1
 |
 | [3] => stdClass Object
 | |
 | | [key] => cpanel_username
 | | [value] => domainco
 | | [serialized] => 0
 | | [encrypted] => 0
 |
 |
[package_pricing] => stdClass Object
 |
 | [id] => 22
 | [package_id] => 13
 | [term] => 1
 | [period] => month
 | [price] => 8.0000
 | [setup_fee] => 0.0000
 | [cancel_fee] => 0.0000
 | [currency] => USD
 |
[package] => stdClass Object
 |
 | [id] => 13
 | [id_format] => {num}
 | [id_value] => 1
 | [module_id] => 4
 | [name] => Bronze Standard
 | [description] =>
 | [description_html] => <p>
Order Received (Mobile)

The ticket received email template allows for the following tags:

<table>
<thead>
<tr>
<th>Tags</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>(order.id)</td>
<td>The system-level order ID</td>
<td></td>
</tr>
<tr>
<td>(order.number)</td>
<td>The order number</td>
<td>e.g. &quot;519131eb523b8&quot;</td>
</tr>
<tr>
<td>(order.client_id)</td>
<td>The system-level client ID of the client this order belongs to</td>
<td></td>
</tr>
<tr>
<td>(order.client_id_code)</td>
<td>The friendly client ID code</td>
<td></td>
</tr>
<tr>
<td>(order.client_first_name)</td>
<td>The first name of the client</td>
<td></td>
</tr>
<tr>
<td>(order.client_last_name)</td>
<td>The last name of the client</td>
<td></td>
</tr>
<tr>
<td>(order.client_company)</td>
<td>The client's company name</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>order.client_address1</td>
<td>The client's primary address</td>
<td></td>
</tr>
<tr>
<td>order.client_email</td>
<td>The client's email address</td>
<td></td>
</tr>
<tr>
<td>invoice.id</td>
<td>The system-level invoice ID of the invoice created for this order</td>
<td></td>
</tr>
<tr>
<td>invoice.id_code</td>
<td>The friendly invoice ID code</td>
<td></td>
</tr>
<tr>
<td>invoice.date_due</td>
<td>The UTC datetime stamp of the invoice due date</td>
<td></td>
</tr>
<tr>
<td>invoice.id_code</td>
<td>Formatted as “YYYY-MM-DD HH:MM:SS”</td>
<td></td>
</tr>
<tr>
<td>invoice.date_billed</td>
<td>The UTC datetime stamp of the invoice bill date</td>
<td></td>
</tr>
<tr>
<td>invoice.id_code</td>
<td>Formatted as “YYYY-MM-DD HH:MM:SS”</td>
<td></td>
</tr>
<tr>
<td>invoice.total</td>
<td>The total amount due on the invoice</td>
<td></td>
</tr>
<tr>
<td>invoice.paid</td>
<td>e.g. “8.00”</td>
<td></td>
</tr>
<tr>
<td>invoice.due</td>
<td>The remaining amount due</td>
<td></td>
</tr>
<tr>
<td>invoice.currency</td>
<td>e.g. “USD”</td>
<td></td>
</tr>
<tr>
<td>invoice.currency</td>
<td>The currency that the invoice is in</td>
<td></td>
</tr>
</tbody>
</table>

Due to the nature of tag objects containing several fields, many of which are likely irrelevant for use in email templates, but may be useful to you in certain circumstances, a dump of the tags are shown below.

**The {order} object**

```php
stdClass Object
{
    [id] => 52
    [order_number] => 519131eb623b8
    [order_form_id] => 1
    [invoice_id] => 19983
    [fraud_report] =>
    [status] => pending
    [date_added] => 2013-05-13 18:33:15
    [fraud_status] =>
    [order_form_label] => my_order_form
    [order_form_name] => My Order Form
    [client_id] => 3
    [total] => 8.0000
    [paid] => 0.0000
    [currency] => USD
    [date_closed] =>
    [invoice_id_code] => PD-15695
    [client_id_code] => PD-1501
    [client_first_name] => Mark
    [client_last_name] => Stevens
    [client_company] => Comp Co.
    [client_address1] => 123 Test St.
    [client_email] => user@domain.com
    [services] => Array
    {
        [0] => stdClass Object
        {
            [order_id] => 52
            [service_id] => 234
        }
    }
}
```

**The {invoice} object**

```php
stdClass Object
{
    [id] => 19983
    [id_format] => PD-{num}
    [id_value] => 15695
    [client_id] => 3
    [date_billed] => 2013-05-13 18:33:15
    [date_due] => 2013-05-13 18:33:15
```
The `services` array of objects

```php
Array
{
  [0] => stdClass Object
    {
      [id] => 54940
      [invoice_id] => 19983
      [service_id] => 234
      [description] => Bronze Standard - domain.com
      [qty] => 1.0000
      [amount] => 8.0000
      [subtotal] => 8.00000000
      [taxes] => Array
        {
        }
      [taxes_applied] => Array
        {
        }
      [tax_subtotal] => 0
      [tax_total] => 0
      [total] => 8
      [total_w_tax] => 8
    }
  [delivery] => Array
    {
      [0] => stdClass Object
        {
          [id] => 20716
          [invoice_id] => 19983
          [method] => email
          [date_sent] =>
        }
    }
  [meta] => Array
    {
    }
```

[parent_service_id] => 
[package_group_id] => 1
[id_format] => PD-{num}
[id_value] => 229
[pricing_id] => 22
[client_id] => 3
[module_row_id] => 6
[coupon_id] => 
[qty] => 1
[status] => in_review
[date_renews] => 2013-06-13 18:33:14
[date_last_renewed] =>
[date_suspended] =>
[date_canceled] =>
[id_code] => PD-229
[fields] => Array
  |
  [0] => stdClass Object
   |
   [key] => cpanel_confirm_password
   [value] => password
   [serialized] => 0
   [encrypted] => 1
  |
  [1] => stdClass Object
   |
   [key] => cpanel_domain
   [value] => domain.com
   [serialized] => 0
   [encrypted] => 0
  |
  [2] => stdClass Object
   |
   [key] => cpanel_password
   [value] => password
   [serialized] => 0
   [encrypted] => 1
  |
  [3] => stdClass Object
   |
   [key] => cpanel_username
   [value] => domainco
   [serialized] => 0
   [encrypted] => 0
  |

[package_pricing] => stdClass Object
  |
  [id] => 22
  [package_id] => 13
  [term] => 1
  [period] => month
  [price] => 8.0000
  [setup_fee] => 0.0000
  [cancel_fee] => 0.0000
  [currency] => USD
  |

[package] => stdClass Object
  |
  [id] => 13
  [id_format] => {num}
  [id_value] => 1
  [module_id] => 4
  [name] => Bronze Standard
Linking to Order Pages

A link to each order page is displayed in the staff area, under [Packages] > [Order Forms]. It's also possible to link directly to a specific package, pricing option, or step.

Linking to a Specific Order Form

The default order form exists at /order/, relative to your installation path. Other order forms may be accessed through their unique label identifier. Below are a few examples:

<table>
<thead>
<tr>
<th>Name</th>
<th>Label</th>
<th>URI</th>
<th>Default Order Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Design</td>
<td>design</td>
<td>/order/</td>
<td>YES</td>
</tr>
</tbody>
</table>

or
### Linking to a Specific Product

The following examples demonstrate how to link to a particular product or price point.

<table>
<thead>
<tr>
<th>Description</th>
<th>Parameters</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Package Group</td>
<td>• <strong>group_id</strong> The ID of the package group to display.</td>
<td>/order/main/packages/label/?group_id=1</td>
</tr>
<tr>
<td>Display Package Group with Package Selected</td>
<td>• <strong>group_id</strong> The ID of the package group to display.</td>
<td>/order/main/packages/label/?group_id=1&amp;package_id=2</td>
</tr>
<tr>
<td>Display Package Configuration with Price Selected</td>
<td>• <strong>group_id</strong> The ID of the package group to display.</td>
<td>/order/main/packages/label/?group_id=1&amp;pricing_id=5</td>
</tr>
<tr>
<td></td>
<td>• <strong>pricing_id</strong> The ID of the package pricing.</td>
<td>/order/main/packages/label/?group_id=1&amp;pricing_id=5</td>
</tr>
<tr>
<td></td>
<td>• <strong>configoptions</strong> A list of option IDs and their values (since plugin v2.21.0+)</td>
<td>/order/config/index/?group_id=1&amp;pricing_id=5&amp;configoptions[20]=yes&amp;configoptions[7]=default%20option</td>
</tr>
</tbody>
</table>

Note that this is the ID of the package as seen in the URL when editing the package.

#### Order Template

<table>
<thead>
<tr>
<th>Order Template</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>AJAX Boxes</td>
<td>/order/main/packages/label/?group_id=1&amp;pricing_id=5</td>
</tr>
<tr>
<td>AJAX Slider</td>
<td>/order/main/packages/label/?group_id=1&amp;pricing_id=5</td>
</tr>
</tbody>
</table>

**Configurable Options may be set in version 2.21.0+ of the Order plugin**

Specific configurable options can be set by specifying each option in the URI using the option ID and the option value in the format "configoptions[OPTIONID]=VALUE", e.g.:

/\order/config/index/label/?group_id=1&pricing_id=5&configoptions[20]=yes&configoptions[7]=default%20option

Note: Be sure your option values are properly **URL encoded**!
Linking to an Order Form to Include a Coupon

As of version 2.4.0 of the Order plugin, a coupon can be set in the URL to an order form, and will be applied automatically as the user continues through the order process. The coupon may be set automatically when added to a URL at /order/main/ or /order/config/ (in v2.6.0), using the parameter `coupon`. Consider a couple examples:

When including a coupon code in a URL, the coupon code must be URL encoded.

<table>
<thead>
<tr>
<th>Description</th>
<th>Parameters</th>
<th>URI</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set a Coupon to an Order Form</td>
<td>coupon</td>
<td>/order/main/index/label/?coupon=CPN123</td>
<td>In this example, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Coupon code is CPN123</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Order form label is label</td>
</tr>
<tr>
<td>Set a Coupon to a Specific Product</td>
<td>group_id</td>
<td>/order/main/index/vps/?group_id=1&amp;package_id=5&amp;coupon=10off</td>
<td>In this example, the</td>
</tr>
<tr>
<td></td>
<td>package_id</td>
<td></td>
<td>• Coupon code is 10off</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Order form label is vps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Package Group ID is 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Package ID is 5</td>
</tr>
<tr>
<td></td>
<td>coupon</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>configoptions</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>Order Template</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AJAX Boxes</td>
<td>/order/main/packages/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>AJAX Slider</td>
<td>/order/main/packages/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>AJAX List</td>
<td>/order/main/packages/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>Wizard Boxes</td>
<td>/order/config/index/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>Wizard Slider</td>
<td>/order/config/index/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>Wizard List</td>
<td>/order/config/index/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
<tr>
<td></td>
<td>Standard</td>
<td>/order/config/index/label/?</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>group_id=1&amp;pricing_id=15&amp;coupon=10off&amp;configoptions=20=yes&amp;configoptions=7=default%20option</td>
<td>Note: Be sure your option values are properly URL encoded!</td>
</tr>
</tbody>
</table>

Linking to an Order Form to Include the Currency

If the order form supports more than 1 currency, the currency can be specified in the URL. Consider a couple examples:

<table>
<thead>
<tr>
<th>Description</th>
<th>Parameters</th>
<th>URI</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set the currency for the order form</td>
<td>currency</td>
<td>/order/main/index/label/?currency=USD</td>
<td>In this example, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Currency is USD</td>
</tr>
<tr>
<td>Set the currency to a Specific Product</td>
<td>group_id</td>
<td>/order/main/index/vps/?group_id=1&amp;package_id=5&amp;currency=USD</td>
<td>In this example, the</td>
</tr>
<tr>
<td></td>
<td>package_id</td>
<td></td>
<td>• Currency is USD</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Order form label is vps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Package Group ID is 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Package ID is 5</td>
</tr>
</tbody>
</table>
Linking to an Order Form to Include a Specific Language

If Blesta is configured to support more than one language, the language can be specified in the URL. Consider a couple examples:

<table>
<thead>
<tr>
<th>Description</th>
<th>Parameters</th>
<th>URI</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set the language for the</td>
<td>• lang The</td>
<td>/order/main/index/index/?lang=fr</td>
<td>In this example, the</td>
</tr>
<tr>
<td>order form.</td>
<td>language</td>
<td>_fr</td>
<td>Language is fr _fr</td>
</tr>
</tbody>
</table>

Activating the Affiliate System

As of Blesta 4.11, the Order Manager includes an Affiliate System. To activate the Affiliate System, visit Clients > Affiliates and click the "Enable Affiliate System" button to activate.

Once the Affiliate System is active, you will be taken to the General Settings. It’s important to set these details now. When a client signs up to become an affiliate, their settings will initially mirror these settings, but can be overridden on an individual basis as desired.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order Frequency</td>
<td>First</td>
<td>(First, Any) First to pay referrals only for the first order a client makes using a referral link, or Any to pay referrals for each order a client makes using a referral link.</td>
</tr>
<tr>
<td>Order Recurring</td>
<td>(checked)</td>
<td></td>
</tr>
</tbody>
</table>
Whether referrals will be paid for services ordered that renew. Uncheck to apply to the initial purchase only.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maturity Days</td>
<td>90</td>
</tr>
<tr>
<td>Cookie Duration (in days)</td>
<td>180</td>
</tr>
<tr>
<td>Commission Type</td>
<td>Percentage</td>
</tr>
<tr>
<td>Minimum Withdrawal Amount</td>
<td>10</td>
</tr>
<tr>
<td>Maximum Withdrawal Amount</td>
<td>100</td>
</tr>
<tr>
<td>Withdrawal Currency</td>
<td>USD</td>
</tr>
<tr>
<td>Signup Content</td>
<td></td>
</tr>
</tbody>
</table>

You can add payout payment methods under Clients > Affiliates > Settings > Payment Methods.

When an affiliate makes a payout request, it can be viewed under Clients > Affiliates > Payouts. Staff can receive an email when an affiliate submits a payout request by subscribing to these notices within the Cog/Gear icon in the Orders widget, which can be found either under Billing > Overview, or the Dashboard.

A listing of all affiliates can be found under Clients > Affiliates > Affiliates.

A list of packages to be excluded from payout calculations can be found under Clients > Affiliates > Settings > Excluded Packages. Check the box for a package to exclude it.
Clients can sign up to be affiliates by clicking the "Affiliates" link in the client area. The client will see the "Signup Content" as defined above when they visit this page, and will be able to sign up as an affiliate.

Staff can override the affiliates settings by clicking the "Manage Affiliate" link on the client's profile under Actions, as well as see graph statistics.
Common Issues

- reCaptcha always returns "The captcha entered was invalid. Please try again."
  - This may be a result of allow_url_fopen being disabled in your PHP config. To confirm, check your Blesta logs for:
    ```
    general.WARNING:
    E_WARNING: file_get_contents(): https:// wrapper is disabled in the server configuration by allow_url_fopen=0
    ```
    This directive must be changed for reCaptcha to work properly.
  - When a user visits the payment page to pay with PayPal or another non-merchant gateway, they are redirected after 5 seconds without clicking the payment button.
    - This redirect is intentional, but if you are using PayPal Subscriptions and there are 2 buttons, users may not have enough time to make a selection before being redirected. To change the time to redirect, edit `/plugins/order/views/templates/wizard/ajax/javascript/checkout_complete.js` and change the value for delay(5000) to the desired value. The number represents milliseconds.