



Red Hat 3scale 2-saas

Billing

For Use with Red Hat 3scale 2-saas

Red Hat 3scale 2-saas Billing

For Use with Red Hat 3scale 2-saas

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This guide documents billing management with Red Hat 3scale 2-saas.

Table of Contents

CHAPTER 1. CONFIGURE BILLING	3
CHAPTER 2. ADDING VAT RATE ON BILLING	4
CHAPTER 3. CREDIT CARD FLOW	6
3.1. PREREQUISITES	6
3.2. STEP 1: HIDE CERTAIN MENU ITEMS WHEN CREDIT CARD DETAILS ARE MISSING	6
3.2.1. Editing users_menu partial	6
3.2.2. Editing submenu partial	7
3.3. STEP 2: ENABLE FEATURE	9
3.4. STEP 4: ENSURE CREDIT CARD IS ENTERED BY DEVELOPER WHEN UPGRADING FROM A FREE TO A PAID PLAN	9
3.4.1. Editing applications/show Template	9
3.4.2. Editing applications/form partial	9
CHAPTER 4. STRIPE INTEGRATION (RECOMMENDED)	10
4.1. PREREQUISITES	10
4.2. STEP 1: GET YOUR API KEYS FROM STRIPE	10
4.3. CONFIGURE SETTINGS IN 3SCALE	10
4.3.1. Note	13
CHAPTER 5. ADYEN INTEGRATION	14
5.1. PREREQUISITES	14
5.2. STEP 1: FIND THE INFORMATION IN YOUR ADYEN ACCOUNT	14
5.2.1. Step 2: Configure your payment gateway in your 3scale account	16
5.2.2. Step 3: Enable the "alias" additional data in the Adyen API response	17
5.2.3. Step 4: Test your billing workflow	17
5.2.4. Troubleshooting	18
CHAPTER 6. BRAINTREE INTEGRATION	19
6.1. STEP 1: GET YOUR API KEYS FROM BRAINTREE	19
6.2. STEP 2: CONFIGURE YOUR SETTINGS IN 3SCALE	20
6.2.1. Note	22
6.2.2. Troubleshooting	22
CHAPTER 7. OGONE INTEGRATION	23
7.1. STEP 1: GET YOUR API KEYS FROM OGONE	23
7.2. STEP 2: CONFIGURE YOUR SETTINGS IN 3SCALE	25
7.2.1. Note	27
7.2.2. Troubleshooting	27
CHAPTER 8. AUTHORIZE.NET INTEGRATION	29

CHAPTER 1. CONFIGURE BILLING

In this tutorial you'll learn how the 3scale billing system works and how to set it up.

When setting up your billing configuration, you can choose between Prepaid and Postpaid.

- **Prepaid mode:** In prepaid mode, all fixed fees and setup fees are billed immediately at the beginning of the month or at the beginning of any pro-rated billing period. Variable costs are always calculated at the end of the month.
- **Postpaid mode:** In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

To do so, go to **Audience > Billing > Charging & Gateway**.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:

Postpaid

In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☐ **Charging enabled**

Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will be issued, but no real transaction will take place.

Currency

USD - American Dollar ▾

Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

Invoice footnote

Text to show if VAT/Sales Tax is 0%

Billing periods for invoice ids

monthly ▾

[Save](#)

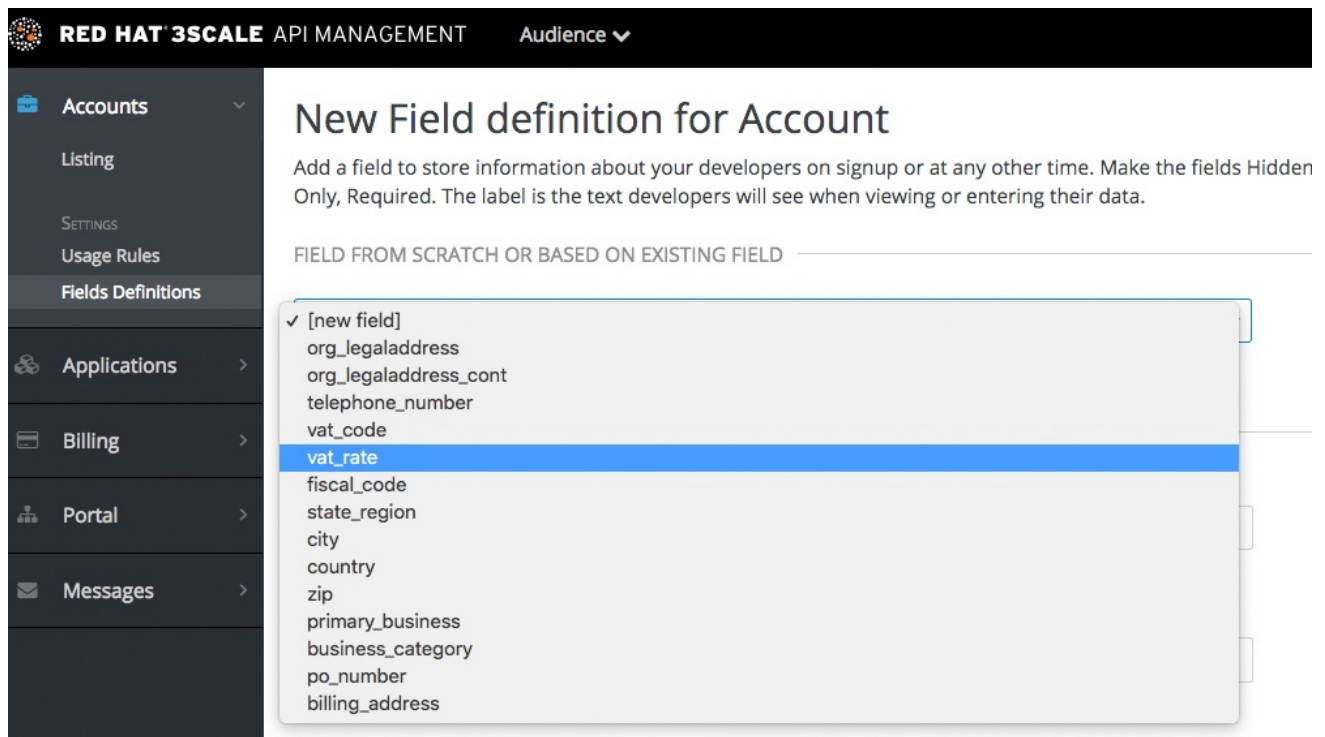
All paid plans are charged monthly.

CHAPTER 2. ADDING VAT RATE ON BILLING

API providers who charge their customers for using their API service often need to charge the tax as well. This article will explain how 3scale billing system supports VAT.

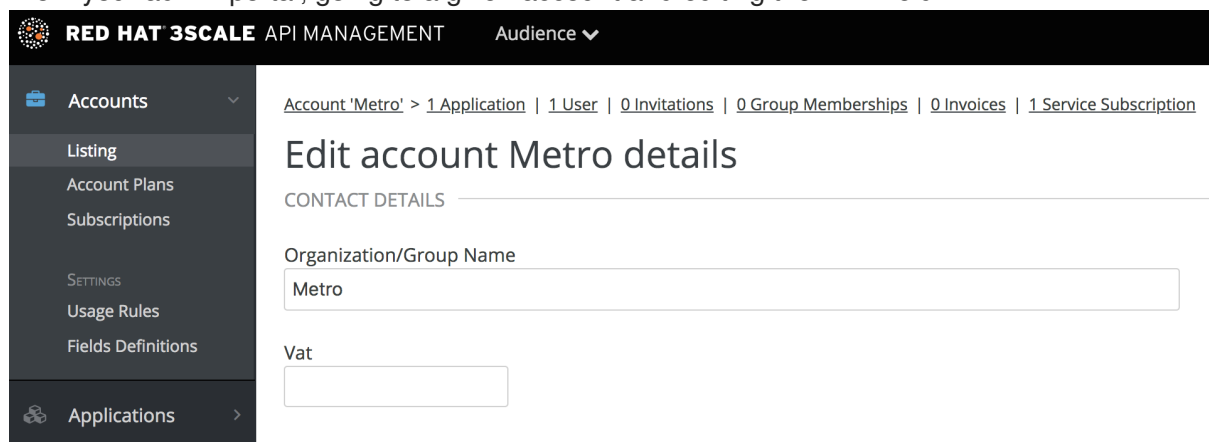
If you have Billing feature enabled, the steps to add VAT rates are as follows:

1. From the Admin Portal, go to **Audience > Accounts > Fields Definitions**
2. Then in the "Account" object click "Create" on the right hand side to add a field
3. From the "new field" drop down select "vat_rate"



Either you can allow developers to set their own VAT rate, or alternatively you can set the VAT rate for them. If you choose the latter you have the following options:

- From your admin portal, going to a given account and editing the VAT field



- From the Account Management API, using the Account Update endpoint. If you want to update several accounts you will have to get them first using the Account List endpoint and then iterate over them updating the field. This could be combined into an automated flow using Webhooks to trigger your update requests when the Account is created or modified.

VAT rate will be calculated and reflected in invoices.

CHAPTER 3. CREDIT CARD FLOW

This guide will help you define a great workflow in your Developer Portal to ensure that developers submit their credit card details before they gain access to your paid API(s).

It's increasingly common to see businesses charge for access to their APIs. Sometimes the entire business model of a company is built around paid APIs, while other times it's an important revenue stream to cover costs or generate profits for the company.

3.1. PREREQUISITES

- 3scale finance module (billing and charging) activated and configured
- Paid plans defined (usually application plans)

After following the steps detailed below, all the existing developers who have paid plans but no credit card details entered, will be redirected to the credit card details form. They will not be allowed to access the portal until they have successfully entered their credit card details.

3.2. STEP 1: HIDE CERTAIN MENU ITEMS WHEN CREDIT CARD DETAILS ARE MISSING

When a developer signs up to a paid plan and is required to fill in the credit card details, it is recommended to hide some menu items since it will not be possible to navigate there until the credit card is entered. In order to do so, you will need to change the following 2 menu partials:

- users_menu
- submenu

3.2.1. Editing users_menu partial

You will need to replace the whole of the partial content with the below:

```
<ul class="nav nav-tabs">
  {% unless current_account.requires_credit_card_now? %}
    <li class="{% if urls.account_overview.active? %}active{% endif %}">
      <a href="{% urls.account_overview %}">Account Details</a>
    </li>

    {% if provider.account_management_enabled? %}
      {% include 'menu_item' with urls.users %}

      {% if provider.multiple_users_allowed? %}
        {% include 'menu_item' with urls.invitations %}
      {% endif %}
    {% endif %}

    {% if provider.account_plans.size > 1 %}
      {% include 'menu_item' with urls.account_plans %}
    {% endif %}
  {% endunless %}

  {% if provider.finance_allowed? %}
```

```

    {% unless current_account.requires_credit_card_now? %}
      {% include 'menu_item' with urls.invoices %}
    {% endunless %}

    {% if urls.payment_details %}
      {% include 'menu_item' with urls.payment_details %}
    {% endif %}
  {% endif %}
</ul>

```

If you have customised this partial already, e.g by adding some css, modifying the HTML tags, etc... you should use the new liquid tag for this purpose accordingly:

```
current_account.requires_credit_card_now?
```

3.2.2. Editing submenu partial

The developer gains access to their keys to start making calls on your API through the API credentials. The changes below ensure that you only display the links if the account does not require credit card details to be entered e.g because there are no paid plans or if the credit card is already available. If you wanted to be completely strict about this, you would edit the system page for displaying the keys to restrict the display – but this would be an advanced customization. You may also want to delete the snippet to display credentials directly on the home page after login.

You will need to replace the whole of the partial content with the below:

```

{% if current_user %}
  {% unless current_account.requires_credit_card_now? %}
    <ul class="nav navbar-nav">

      {% if provider.multiple_applications_allowed? %}
        <li class="{% if urls.applications.active? %}active{% endif
%}">
          <a href="{{ urls.applications }}">{{
urls.applications.title }}</a>
        </li>
        {% elsif current_account.applications.first == present%}
          {% assign app = current_account.applications.first %}
          <li class="{% if app.url.active? %}active{% endif %}">
            <a href="{{ app.url }}">API Credentials</a>
          </li>
          {% elsif current_user.can.create_application? %}
            <li class="{% if url.new_application.active? %}active{% endif
%}">
              <a href="{{ urls.new_application }}">Get API Credentials</a>
            </li>
          {% else %}
            <!-- You don't have any application neither can create one.
Bad luck. -->
          {% endif %}

          {% assign live_apps_present = current_account.applications |
map: 'state' | any: 'live' %}
          {% if live_apps_present %}
            <li class="{% if urls.stats.active? %}active{% endif %}">

```

```

        <a href="{{ urls.stats }}">Statistics</a>
      </li>
    {% endif %}

    {% if provider.multiple_services_allowed? %}
      <li class="{% if urls.services.active? %}active{% endif %}">
        <a href="{{ urls.services }}">{{ urls.services.title }}</a>
      </li>
    {% endif %}

    <li><a class="{% if urls.docs.active? %}active{% endif %}"
href="/docs">Documentation</a></li>

  </ul>
{% endunless %}
<ul id="user_widget" class="nav navbar-nav navbar-right">
  {% unless current_account.requires_credit_card_now? %}
    <li class="{% if urls.messages_inbox.active? %}active{% endif
%}">
      <a href="{{ urls.messages_inbox }}">
        Messages
        {% if current_account.unread_messages.size > 0 %}
          {{ current_account.unread_messages.size }}
        {% endif %}
      </a>
    </li>
    <li class="{% if urls.account_overview.active? %}active{% endif
%}">
      <a href="{{ urls.account_overview }}">

        Settings
      </a>
    </li>
  {% endunless %}
  <li>
    <a id="sign-out-button" class="pull-right sign-out"
title="Logout {{ current_user.username }}" href="{{ urls.logout }}">

      </a>
    </li>
  </ul>

{% else %}
  <ul class="nav navbar-nav">
    <li><a href="/docs">Documentation</a></li>
    <li><a href="/#plans">Plans</a></li>
  </ul>
  <ul class="nav navbar-nav navbar-right">
    <li>
      <a href="{{ urls.login }}">
        Sign in
      </a>
    </li>
  </ul>
{% endif %}

```

Again, if you have customised this partial already, you will need to modify your version accordingly using the same new liquid tag: `current_account.requires_credit_card_now?`

3.3. STEP 2: ENABLE FEATURE

To enable this feature, you must enable the following 2 switches:

Switches can be found in **Audience > Developer Portal > Feature Visibility**.

- Finance
- Credit Card on Signup (only available if you are on an older 3scale account.)

3.4. STEP 4: ENSURE CREDIT CARD IS ENTERED BY DEVELOPER WHEN UPGRADING FROM A FREE TO A PAID PLAN

You may allow developers to select plans when changing plans. The ideal setting is Request credit card entry for paid plans. This will ensure that if a paid plan is selected, the plan change can only be done instantly if a credit card is stored. Otherwise a notification is displayed to point the developer to the location to enter their card details.

This setting is defined from the Admin Portal. Go to **[your_API_name] > Integration > Settings > Application Plans**. Here you can select the option Request credit card entry for paid plans in the section *Application Plans* of the settings form.

To enable this behavior, you will need to make changes to the following CMS pages:

- applications/show template
- applications/form partial

3.4.1. Editing applications/show Template

You will need to look for `plan_widget` in the page code and change it to the following:

```
{% plan_widget application, wizard: true %}
```

If you have added the `plan_widget` in any other pages, you will need to update those to match the above snippet.

3.4.2. Editing applications/form partial

Follow the same instructions above to change any instances of `plan_widget` in the applications/form partial to conform to the new snippet.

And that's it! You can now enforce the addition of credit card details before allowing usage of your API for paid plans.

CHAPTER 4. STRIPE INTEGRATION (RECOMMENDED)

After completing these steps you will have configured Stripe as a payment gateway for your account. This will allow your developers to enter their credit card details, and you can automatically charge them through Stripe for access to your API, according to the calculated invoices.

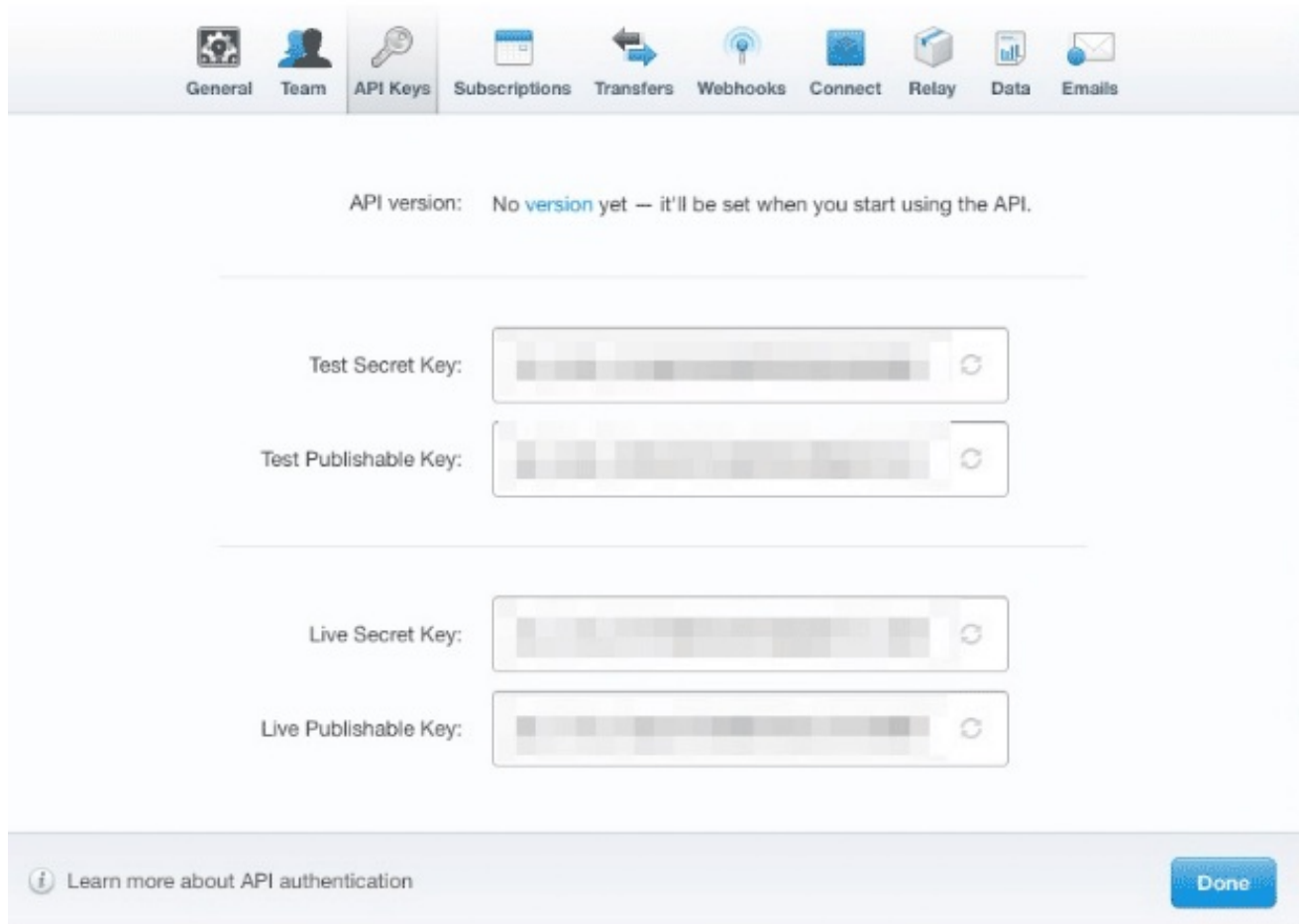
If you enable credit card charging for your paid API, then one key step is to setup your [payment gateway](#). There are a number of alternative payment gateways which you can use with your 3scale account. Here we cover the steps for Stripe.

4.1. PREREQUISITES

Before you start these steps, you will need to open an account with [Stripe](#).

4.2. STEP 1: GET YOUR API KEYS FROM STRIPE

Log in to your Stripe account, and get your API keys at <https://dashboard.stripe.com/account/apikeys>. You will need two keys: a "secret" one, and a "publishable" one. Use the "test" set when you're doing tests and the "live" ones when you are ready to start charging.

The screenshot shows the Stripe dashboard's 'API Keys' section. At the top, there's a navigation bar with icons for General, Team, API Keys (selected), Subscriptions, Transfers, Webhooks, Connect, Relay, Data, and Emails. Below the navigation bar, the 'API version' is listed as 'No version yet — it'll be set when you start using the API.' The main content area contains four input fields for API keys, each with a refresh icon (circular arrow) to its right. The first two fields are for 'Test' keys: 'Test Secret Key' and 'Test Publishable Key'. The next two fields are for 'Live' keys: 'Live Secret Key' and 'Live Publishable Key'. All fields contain masked characters. At the bottom left, there is a link 'Learn more about API authentication' with an information icon. At the bottom right, there is a blue 'Done' button.

4.3. CONFIGURE SETTINGS IN 3SCALE

You need to tell 3scale to start using these API keys. To do this, log in to your 3scale admin portal and go to **Audience > Billing > Charging & Gateway**.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid

In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☐ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar

Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

Invoice footnote

Text to show if VAT/Sales Tax is 0%

Billing periods for invoice ids
monthly

[Save](#)

If the Charging Enabled flag is not active, enable it and click Save.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid

In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☒ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar

Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

Invoice footnote

Text to show if VAT/Sales Tax is 0%

Billing periods for invoice ids
monthly

[Save](#)

You should see a drop-down called Gateway near the bottom of the page. Change it to Stripe.



NOTE

Integration with Authorize.Net and Ogone is no longer supported.

Credit card gateway

Select your credit card gateway and configure it with the details of your merchant account.

BASICS



Gateway

Stripe

The form below the drop-down should change to show two fields. Insert your Stripe API keys and click Save.

Credit card gateway

Select your credit card gateway and configure it with the details of your merchant account.

BASICS



Gateway

Stripe

STRIPE OPTIONS

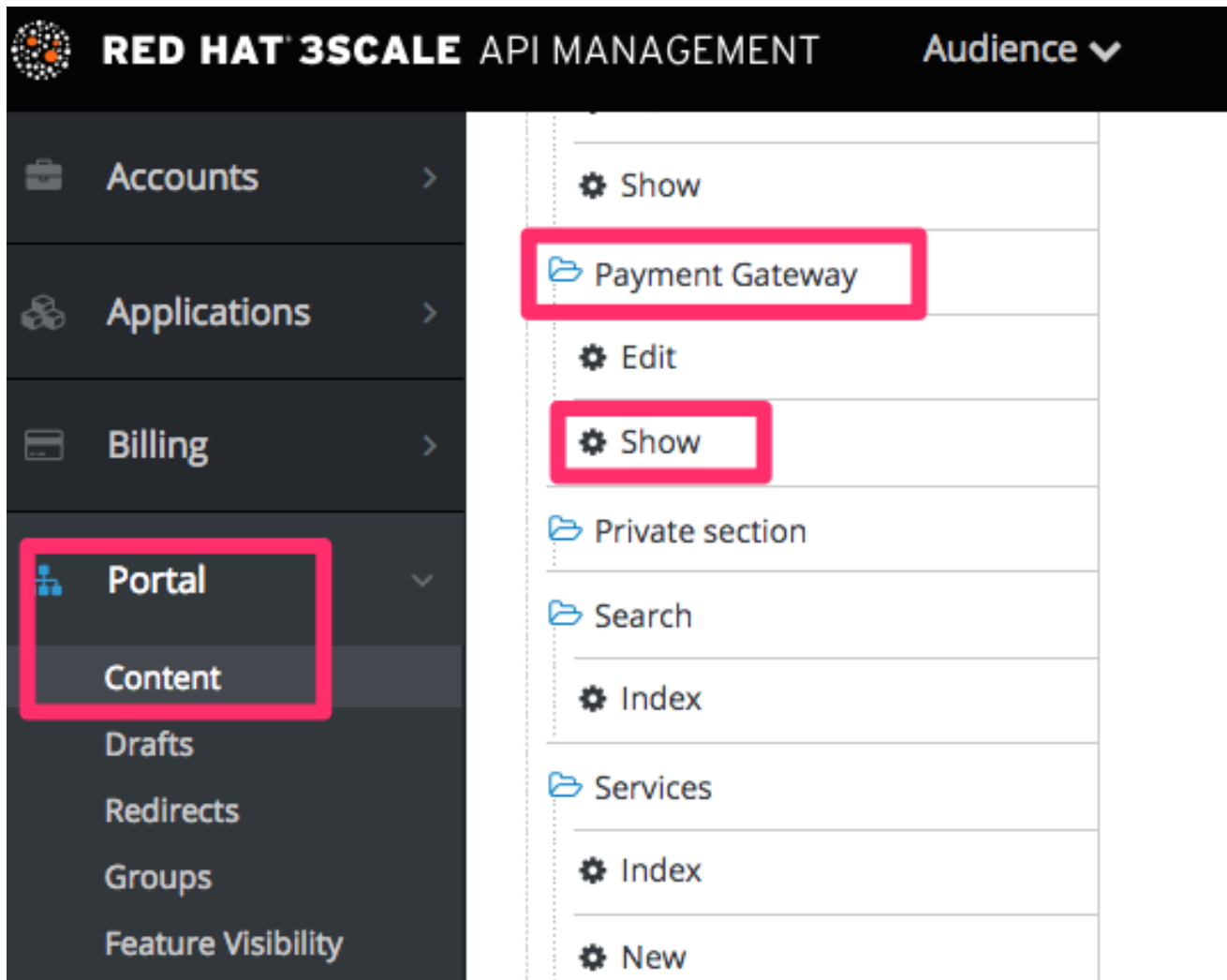


Secret Key

Publishable Key

You might see a couple of alerts when you change your payment gateway. This is expected. Read them and accept them if they appear.

The payment gateway is now set up, but your users might not be able to use it yet since it's not configured in the CMS. Go to Developer portal, and click on the template called Payment Gateway / Show on the left navigation pane.



If it's not there already, add the following code before `{% when "braintree_blue" %}`:

```
{% when "stripe" %}
  <p><a href="{{ current_account.edit_stripe_billing_address_url }}">Edit
  billing address</a></p>

  {% if current_account.has_billing_address? %}
    {% stripe_form "Edit Credit Card Details" %}
  {% else %}
    <p>After entering billing address, the option to enter credit card will
    be enabled.</p>
  {% endif %}
```

Finally click Save and Publish. Your users should now be able to pay you using the Stripe gateway.

4.3.1. Note

In order to map your data from Stripe with your data on 3scale, you can use the Stripe field called `metadata.3scale_account_reference` which is composed of `3scale-[PROVIDER_ID]-[DEVELOPER_ACCOUNT_ID]`

CHAPTER 5. ADYEN INTEGRATION

After completing these steps, you'll have configured Adyen as a payment gateway for your account. This will allow your developers to enter their credit card details, and you can automatically charge them through Adyen for access to your API, according to the calculated invoices.

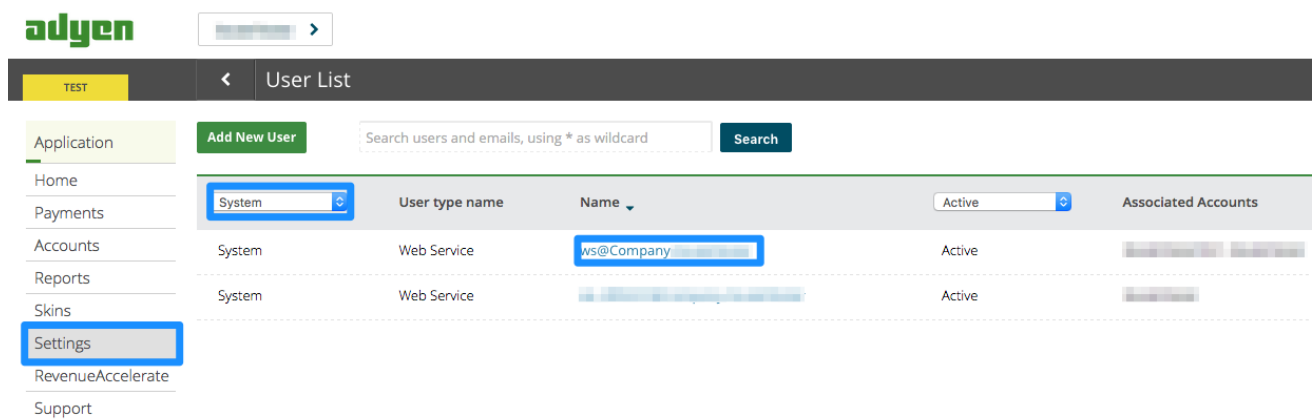
Setting up your payment gateway is a key step enabling credit card charging for use of your paid API. There are a number of alternative payment gateways you can use with your 3scale account. Here we cover the steps for Adyen.

5.1. PREREQUISITES

Before you start these steps, you'll need to open an account with [Adyen](#). You need a *Company* account and a *Merchant* account within it (sub-account). There are a number of requirements that must be fulfilled before you can apply for a live account with Adyen. You can see what those requirements are [here](#).

5.2. STEP 1: FIND THE INFORMATION IN YOUR ADYEN ACCOUNT

To begin, log into your Adyen account. Then find your credentials in the **Settings > Users** area, and select *system* from the dropdown menu in the following view:



The screenshot shows the Adyen user management interface. On the left, a sidebar contains menu items: Application, Home, Payments, Accounts, Reports, Skins, Settings (highlighted with a blue box), RevenueAccelerate, and Support. The main area is titled 'User List' and includes an 'Add New User' button and a search bar. Below the search bar is a table with columns: System (a dropdown menu with 'System' selected), User type name, Name, Active (a dropdown menu with 'Active' selected), and Associated Accounts. The table contains two rows. The first row shows 'System' as the user type, 'Web Service' as the user type name, 'ws@Company' as the name (highlighted with a blue box), 'Active' as the status, and a list of associated accounts. The second row shows 'System' as the user type, 'Web Service' as the user type name, a redacted name, 'Active' as the status, and a list of associated accounts.

System	User type name	Name	Active	Associated Accounts
System	Web Service	ws@Company	Active	[redacted]
System	Web Service	[redacted]	Active	[redacted]

Click on the "Company" account (the account at the top of the list). You will then be taken to the Company account settings view:

adyen 3scaleTester >

TEST < Edit Web Service User

Application

- Home
- Payments
- Accounts
- Reports
- Skins
- Settings**
- RevenueAccelerate
- Support

User Account Details

User Name **Login**

Last Password Change 2016-05-05 15:39:51 CEST

Password **Secret Password**

Retype Password

Generate Password **Generate POS Password**

Client Certificate (DN)

Easy Encryption

Public Key

Client Encryption Public Key

Preformatted for direct insertion in your code:

"10001|C9CEA1701A8BB203254B314A081C814E8..."

Hosted Client Side Encryption

Library Version Always latest version

Library location applied after you click save)

Library location

The V0_1_X CSE library is available, running library version 0_1_17.

Roles and Associated Accounts

Roles Accounts

Type to search associate accounts Search

3scaleTester Company **Merchant Id**

MerchantAccount

Edit Allowed User IP Range

Add IP address 0 . 0 . 0 . 1

Allow 32 (this IP address only)

Allow

No allowed IP address ranges configured

Save

Finally you have access to the **Login**, **Secret Password**, **Client Encryption Public Key**, **Merchant ID** and **Library location** that you will need for the 3scale billing settings. To be able to view the **Public Key**, you must click on "generate password" and copy this password somewhere.

Application

- Home
- Payments
- Accounts
- Reports
- Skins
- Settings**
- RevenueAccelerate
- Support

TEST < Edit Web Service User

User Account Details

User Name: [redacted]

Last Password Change: 2016-05-05 15:39:51 CEST

Password: [redacted]

Retype Password: [redacted]

Generate Password **Generate POS Password**

Client Certificate (DN): [redacted]

Easy Encryption

Client Encryption Public Key: [redacted]

Preformatted for direct insertion in your code: [redacted]

Hosted Client Side Encryption

Library Version: **Always latest version** (changes will be applied after you click save)

The V0_1_X CSE library is available, running library version 0_1_17.

Roles and Associated Accounts

Roles Accounts

Type to search roles **Search**

Role	Enabled
API Clientside Encryption Payments role	<input checked="" type="checkbox"/>
iDeal Recurring	<input checked="" type="checkbox"/>
Merchant PAL Webservice role	<input checked="" type="checkbox"/>
Merchant Recurring role	<input checked="" type="checkbox"/>

Edit Allowed User IP Range

Add IP address: 0 . 0 . 0 . 1

Allow: 32 (this IP address only)

Allow

No allowed IP address ranges configured

Save

All these switches should be enabled by default but if for some reason they are not you should enable at least these 3 for successful integration

5.2.1. Step 2: Configure your payment gateway in your 3scale account

In **Audience > Billing > Charging & Gateway**, enable charging by checking the box and click **Save**. You will see all the fields that you need to setup to create the link to your Adyen gateway. Remember to select Adyen from the gateway dropdown menu and save changes.



NOTE

Integration with Authorize.Net and Ogone is no longer supported.

The screenshot shows the Red Hat 3Scale API Management interface. The left sidebar contains navigation links: Accounts, Applications, Billing, Earnings by Month, Invoices, Charging & Gateway (highlighted with a red box), Credit Card Policies, Developer Portal, and Messages. The main content area shows the 'Gateway' dropdown set to 'Adyen' (highlighted with a red box). Below this, the 'ADYEN OPTIONS' section is highlighted with a red box and contains the following fields: Login, Secret Password, Client Encryption Public Key, Merchant ID, and Library location.

5.2.2. Step 3: Enable the "alias" additional data in the Adyen API response

By default when credit card authorization requests are sent from 3scale to Adyen, the returned response does not include the unique identifier of the credit card. To ensure that the correct credit card reference is saved in 3scale and the correct card is charged, this additional data needs to be enabled. In order to do this, you should ask Adyen support to enable the "alias" additional data in the response for the authorization request.

5.2.3. Step 4: Test your billing workflow

Make sure you accelerate the test cycle by enabling *Prepaid Mode* to generate the charge within a day or so. Then choose an existing test account and create an invoice with a line item charge added. Charge the account immediately. This testing approach will incur some minor costs, but it is worth it for the peace of mind that everything works fine, before you have real paying developers using your API.

The payment gateway is now set up, but your users might not be able to use it yet since it is not configured in the CMS. Go to the developer portal tab, and find the template called **Payment Gateway / Show** on the left navigation pane.

If it's not there already, add the following snippet after the block of code beginning with `{% when "stripe" %}`

```
{% when "adyen12" %}
{% if current_account.has_billing_address? %}
  {% adyen12_form %}
{% else %}
  <p><a href="{{ current_account.edit_adyen12_billing_address_url }}">First add a billing address</a></p>
{% endif %}
```



NOTE

- For accounts created before 11th May 2016 you must add the snippet above manually. After said date this will be included in the template by default.
- In order to map your data from Adyen with your data on 3scale, you can use the Adyen field called **shopperReference** which is composed of **3scale-[PROVIDER_ID] - [DEVELOPER_ACCOUNT_ID]**

5.2.4. Troubleshooting

If your account is in sandbox mode and you encounter any problems, you'll have to change it to production.

CHAPTER 6. BRAINTREE INTEGRATION

These are the steps to set up the Braintree gateway in order to to charge for use of your API.

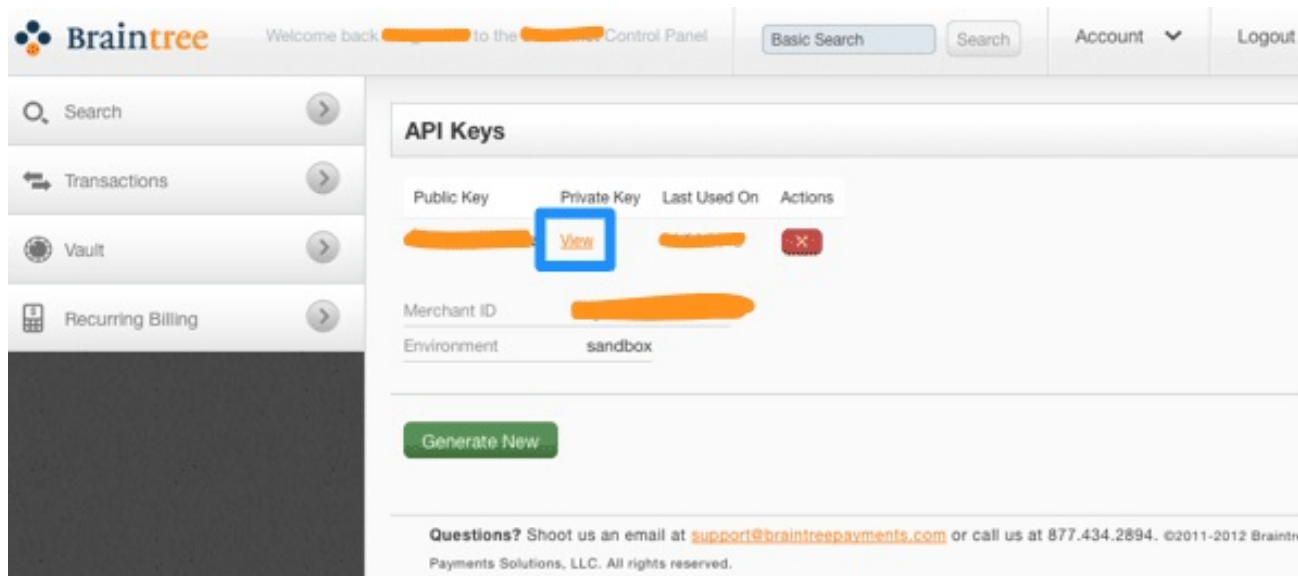
6.1. STEP 1: GET YOUR API KEYS FROM BRAINTREE

You'll need to open an account with [Braintree](#). You need a Gateway and Merchant account plus Vault. As an optional extra, you can choose to allow American Express cards as a payment method.

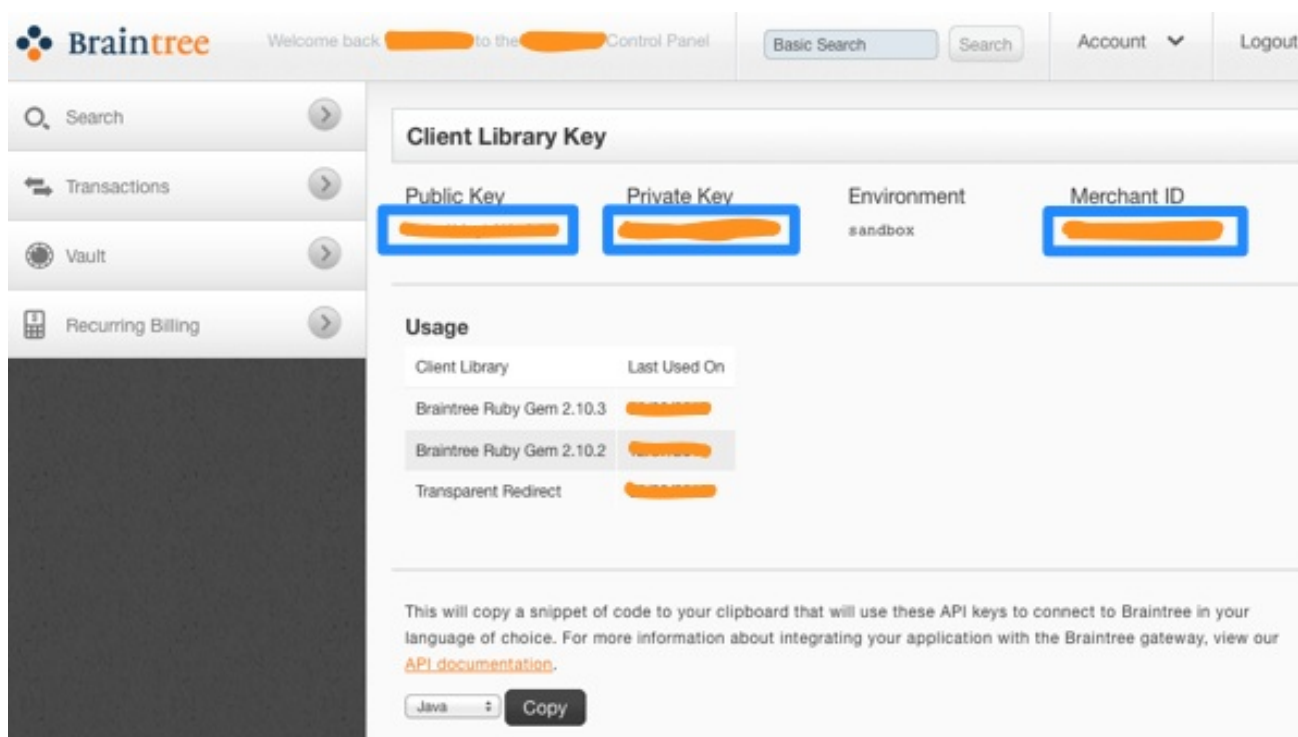
To begin, log in to your Braintree account. Then find your API keys in the **Account > MyUser** area:

The screenshot shows the Braintree Control Panel interface. At the top, the Braintree logo is on the left, and a welcome message "Welcome back [redacted] to the [redacted] Control Panel" is in the center. On the right, there are buttons for "Basic Search", "Search", and a dropdown menu for "Account" which is highlighted with a blue box. A "Logout" link is also present. On the left side, there is a sidebar with navigation links: "Search", "Transactions", "Vault", and "Recurring Billing". The main content area is titled "Edit User test_3scale". It contains several sections: "User Details" with fields for "Name" and "Email" (both redacted); "Change Password (Optional)" with fields for "Current Password", "New Password", and "Confirm New Password"; "Authorization" with a link for "API Keys" highlighted by a blue box; and "Notifications" with a checkbox for "Email Daily Settlement Batch Summary". At the bottom of the form are "Save" and "Cancel" buttons. A footer at the very bottom contains contact information and a copyright notice: "Questions? Shoot us an email at support@braintreepayments.com or call us at 877.434.2894. ©2011-2012 Braintree Payments Solutions, LLC. All rights reserved."

The API keys page still hides the private key, so select the option to view it:



Finally you have the Public Key, Private Key, and Merchant ID that you'll need for the 3scale billing settings:



6.2. STEP 2: CONFIGURE YOUR SETTINGS IN 3SCALE

You need to tell 3scale to start using these API keys. To do this, log in to your 3scale Admin Portal and go to **Audience > Billing > Charging & Gateway**.

Ensure that the currency type specified on the billing page matches the currency type used in your Braintree merchant account.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid

In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☐ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar

Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

Invoice footnote

Text to show if VAT/Sales Tax is 0%

Billing periods for invoice ids
monthly

[Save](#)

If the Charging Enabled flag is not active, enable it and click Save.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid

In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☒ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar

Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

[Save](#)

You should see a drop-down called Gateway near the bottom of the page. Change it to Braintree (Blue Platform).

**NOTE**

Integration with Authorize.Net and Ogone is no longer supported.

Credit card gateway

Select your credit card gateway and configure it with the details of your merchant account.

BASICS

Gateway
Braintree (Blue Platform) ▼

The form below the drop-down should change to show two fields. Insert your Braintree keys and click Save.

BRAINTREE (BLUE PLATFORM) OPTIONS

Public Key

Merchant ID

Private Key

You might see a couple of alerts when you change your payment gateway. This is expected. Read and accept them if they appear.

Your users should now be able to pay you using the Braintree gateway.

6.2.1. Note

In order to map your data from Braintree with your data on 3scale, you can use the Braintree field called **customer.id** which is composed of **3scale-[PROVIDER_ID]-[DEVELOPER_ACCOUNT_ID]**

6.2.2. Troubleshooting

In case your account is in the sandbox mode and you encounter any problems, you will have to change it to production.

CHAPTER 7. OGONE INTEGRATION



WARNING

Ogone integration is deprecated. New integrations are not supported. For integrations existing before 27th of July 2018, Red Hat provides support, but recommends to migrate to one of the fully supported payment gateways: Stripe, Braintree or Adyen.

These are the steps to set up the Ogone gateway in order to to charge for use of your API.

7.1. STEP 1: GET YOUR API KEYS FROM OGONE

You'll need to open an account with Ogone. You need a premium [Ogone](#) ecommerce account – the Horizon activated with Alias Manager option. It's a paid option, which can be activated by Ogone or by Merchant through **backoffice > options > your options**.

The PSPID is used to log into your Ogone account. Then you can find the UserID under **Configuration > Users**.

The screenshot shows the Ogone payment services backoffice interface. The sidebar on the left has 'Configuration' and 'Users' highlighted. The main area is titled 'Users Management' and displays a table of users. A red box highlights the 'PSPID' field in the header, with a red arrow pointing to it and the text 'PSPID from Ogone'. Another red box highlights the 'test3scale' UserID in the first row of the table, with a green arrow pointing to it and the text 'UserID from Ogone'.

UserID	Status	Profile	Scope	
test3scale	Active	Admin	Account	Edit

1 User of 2 Allowed

Make sure that the technical settings of your Ogone account are active. On the Transaction Feedback page, you'll find the SHA-OUT passphrase. On this page you should also ensure that the two settings have checkboxes marked as shown:

ogone payment services [Test] EN FR NL IT DE HOME | SUPPORT

Configuration (highlighted)
Account
Payment methods
Users
Support
Technical information (highlighted)
Advanced
Operations

Explanation of your menu
Userid : test3scale
PSPID : test3scale
Profile : Admin

Transaction feedback (highlighted)

e-Commerce

HTTP redirection in the browser

URL of the web page to be displayed to the customer after the payment process:

Accepturl: displayed when the payment has been authorized, stored, accepted or is waiting to be accepted.
Declineurl: displayed when the acquirer declines the authorization more than the maximum permissible number of times (as defined in the payment retry section of the Global transaction parameters tab).
Exceptionurl: displayed when the payment result is uncertain.
Cancelurl: displayed when the customer cancels the payment.

You can override the default redirection URLs configured above by sending along redirection URLs in the transaction details.

☒ I want to receive transaction feedback parameters on the redirection URLs. (highlighted)

☒ I want Ogone to display a short text to the customer on the secure payment page if a redirection to my website is detected immediately after the payment process. (highlighted)

Direct HTTP server-to-server request

Timing of the request

- ☒ No request.
- ☐ Always deferred (not immediately after the payment).
- ☐ Always online (immediately after the payment to allow customisation of the response seen by the customer).
- ☐ Online but switch to a deferred request in intervals when the online requests fail.

Finally the SHA-IN passphrase is found on the Date and origin verification page:

ogone payment services [Test] EN FR NL IT DE HOME | SUPPORT

Configuration (highlighted)
Passports
Account
Payment methods
Users
Support
Technical information (highlighted)
Advanced
Operations

Explanation of your menu
Userid : test3scale
PSPID : test3scale
Profile : Admin

Data and origin verification (highlighted)

Subscription Type
Subscription: Ogone e-Commerce Horizon
ISP: OIT
Subsidiary: Nederland

Active transaction submission modes
eCommerce
eTerminal
Direct Maintenance
Direct Query
Automatic File Upload
Automatic File Download

Other transaction submission modes available in your account
DirectLink

Checks for e-Commerce

URL of the merchant page containing the payment form that will call the page:orderstandard.asp
You may enter several URL, separated by ","

SHA-IN Pass phrase (highlighted)
***** Visible (highlighted) Enter your SHA-IN pass phrase

Checks for Ogone e-Terminal and Ogone Batch (Manual)
When using these transaction submission modes, you have to log into the back-office beforehand.

Checks for Ogone DirectLink and Ogone Batch (Automatic)
When using these transaction submission modes, login details have to be transmitted in the HTTP parameters.

IP address of the server calling the APIs used for any automatic or direct processing.
You may enter several IP addresses, separated by ","

SHA-IN Pass phrase

Go back to 'Payment Page layout' Confirm and go back to 'Homepage' Confirm and go to 'Transaction feedback'

7.2. STEP 2: CONFIGURE YOUR SETTINGS IN 3SCALE

You need to tell 3scale to start using these API keys. To do this, log in to your 3scale Admin Portal and go to **Settings > Billing**.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid
In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☐ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar
Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

Invoice footnote

Text to show if VAT/Sales Tax is 0%

Billing periods for invoice ids
monthly

[Save](#)

If the Charging Enabled flag is not active, enable it and click Save.

RED HAT 3SCALE API MANAGEMENT Audience ▾

Accounts >

Applications >

Billing ▾

Earnings by Month

Invoices

SETTINGS

Charging & Gateway

Credit Card Policies

Portal >

Messages >

Charging & Gateway

Mode:
Postpaid
In postpaid mode, all fixed fees as well as variable fees are billed at the end of the month.

[Switch to PREPAID](#)

☒ Charging enabled
Enables credit card transactions. When this setting is on, all due invoices will be charged using the selected payment gateway. If you leave this setting off, billing will take place and invoices will issued, but no real transaction will take place.

Currency
USD - American Dollar
Please make sure that the selected currency is supported by your payment gateway. If you are missing currency options, don't hesitate to contact us.

You should see a drop-down called Gateway near the bottom of the page. Change it to Ogone.

Credit card gateway

Select your credit card gateway and configure it with the details of your merchant account.



Basics

Gateway

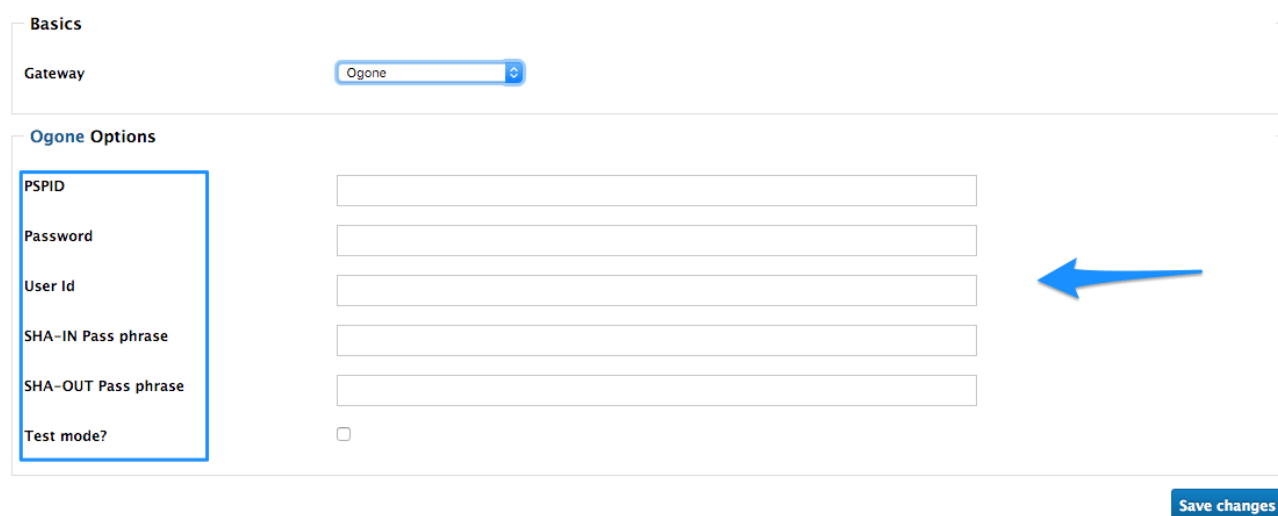
- ✓ Choose one
- Authorize.Net
- Braintree (Blue Platform)
- Ogone
- Stripe

Save changes

The form below the drop-down should change to show two fields. Insert your Ogone API keys and click Save.

Credit card gateway

Select your credit card gateway and configure it with the details of your merchant account.



Basics

Gateway

Ogone

Ogone Options

PSPID

Password

User Id

SHA-IN Pass phrase

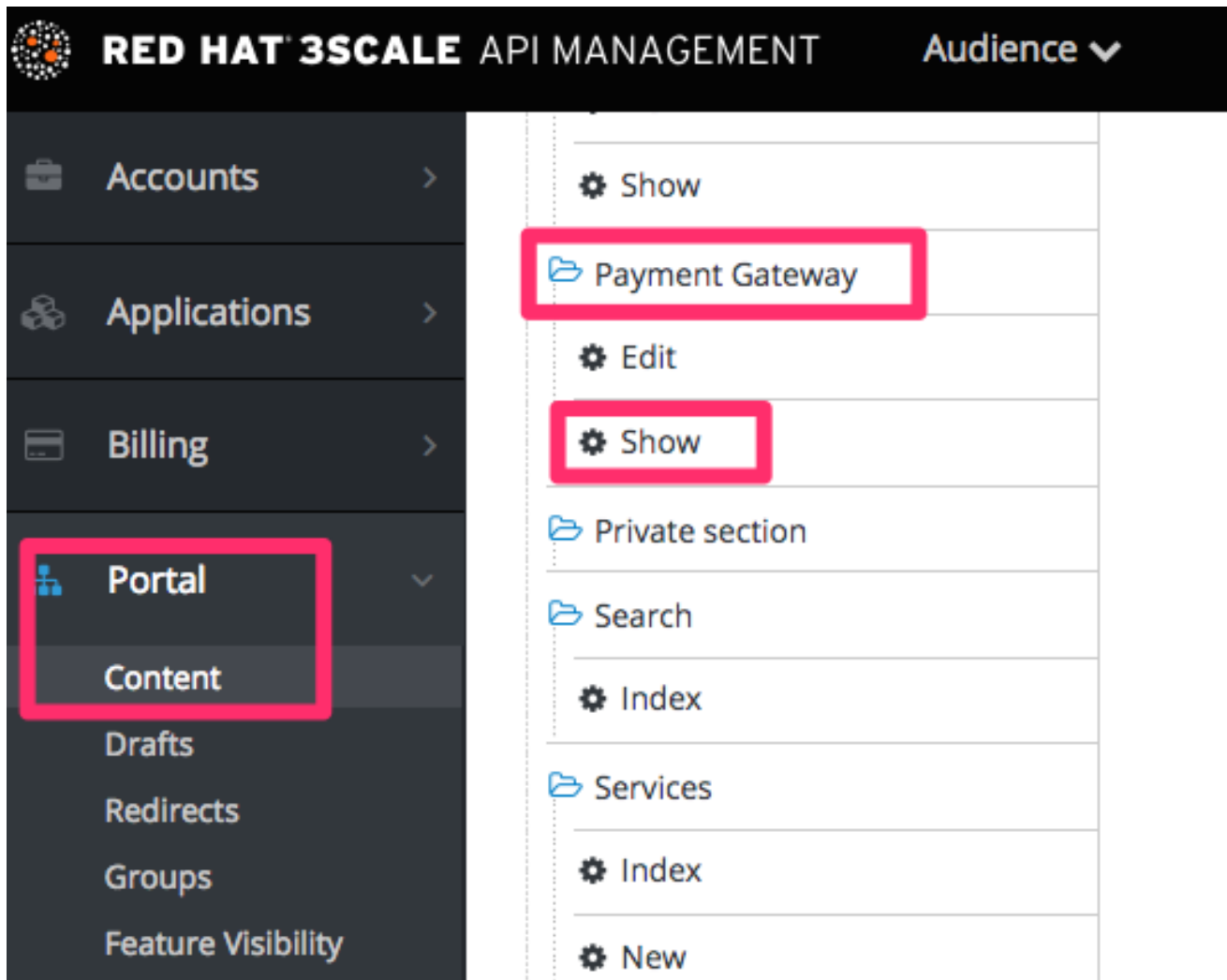
SHA-OUT Pass phrase

Test mode?

Save changes

You might see a couple of alerts when you change your payment gateway. This is expected. Read and accept them if they appear.

The payment gateway is now set up, but your users might not be able to use it yet since it's not configured in the CMS. Go to the Developer Portal page and click on the template called Payment Gateway / Show on the left navigation pane.



If it's not there already, add the following code before `{% when "braintree_blue" %}`:

```
{% when "ogone" %}
  {% if current_account.has_billing_address? %}
    {% if current_account.credit_card_stored? %}
      {% ogone_form "Edit Credit Card Details" %}
    {% else %}
      {% ogone_form "Add Credit Card Details" %}
    {% endif %}
  {% else %}
    <p><a href="{{ current_account.edit_ogone_billing_address_url }}">First add a billing address</a></p>
  {% endif %}
```

Finally click Save and Publish. Your users should now be able to pay using the Ogone gateway.

7.2.1. Note

In order to map your data from Ogone with your data on 3scale, you can use the Ogone field called **alias** which is composed of **3scale-[PROVIDER_ID]-[DEVELOPER_ACCOUNT_ID]**

7.2.2. Troubleshooting

In case something is not working, you can try checking the following tips:

- An overview of all payment methods that work with the alias manager can be found in this document: [Payment methods processing/procedure overview](#).
- In the Ogone admin console go to Technical information → Transaction feedback. Scroll to Directlink/Dynamic Parameters and select all parameters. This will allow you to troubleshoot the error easier (as the error message will be included in the invoice).
- Try to change your ogone password and update this setting in the 3scale console
- Verify if your user was created in your Ogone account with the checkbox "Special user for API (no access to the admin)" checked. You can also search the Ogone Support FAQ for "API user" for the details.
- Make sure that you are entering in 3scale the password of the "API user", and not the password of Ogone PSPID-account
- In case you are using the sandbox mode in Ogone and encounter any errors, change it to production.

CHAPTER 8. AUTHORIZE.NET INTEGRATION



WARNING

Authorize.Net integration is deprecated. New integrations are not supported. For integrations existing before 27th of July 2018, Red Hat provides support, but recommends to migrate to one of the fully supported payment gateways: Stripe, Braintree or Adyen.

Integrate the Authorize.Net payment gateway with the Red Hat 3scale billing system.

Prerequisites:

- A valid Authorize.Net account
- The following Authorize.Net credentials
 - API login ID
 - Transaction key

Configure 3scale API Management:

1. Log in to your 3scale admin portal.
2. Navigate to the **Settings** → **Billing** page.
3. Under the **Basics** section, select **Authorize.Net** from the dropdown menu.
4. Under the **Authorize.Net Options** section, enter your Authorize.Net API login ID and transaction key.
5. Select the **Save Changes** button.