



Payment Integration Manual

<http://www.litepay.com/>

version 1.04

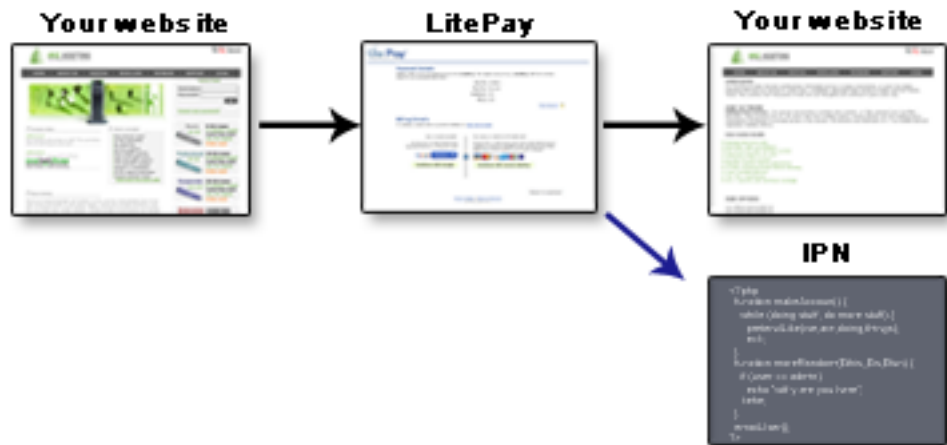
Contents

Introduction	1
Protocal	2
Request	2
IPN Response	2
Implimentation	3
Request front-end	3
Request back-end	3
IPN Response	4

Introduction

The Payment Integration Manual outlines the procedure to integrate the one time payment capture system. This can include payment buttons, encrypted payment buttons, or full use of IPN (Instant Payment Notification -or- API callback).

In a standard environment, the user is transferred from your website, to LitePay's website.



First the user visits your website and either proceeds to checkout with a shopping cart button, or the user clicks a premade payment button.

The user is then transferred to LitePay to enter their details and pay.

After the user has successfully made a payment, they are transferred to your website.

If you have specified an IPN url, or have a default one in your account, LitePay will send the IPN data to your IPN url.

Protocol

Basic Request/Response table and definitions. Please use these as reference to develop your own applications.

Request

This is data the the LitePay system will accept when being pased into the payment system. These values may be pased by GET or POST. Please be aware that invoice ids must be unique. You can not use the same invoice ID twice. Invoice ids may be [0-9a-zA-Z]{1,255}.

Variable	Required	Type	Length	Description
amount	Yes	int	1-255	The price of the one time charge.
business	Yes	text	1-255	Any email address in your LitePay account.
item_name	Yes	text	1-255	Name of the item or product.
item_number	No	int	1-11	Item or product ID number.
image_url	No	text	1-255	URL for logo image at the top on payment page.
invoice	No	text	1-255	Unique Invoice ID for merchant reference and tracking
bn	No	text	1-255	Business name on receipt.
return	Yes	text	1-255	After a successfull transaction the user returns to this URL.
cancel_return	No	text	1-255	When a user cancels they go back to this URL.
notify_url	No	text	1-255	URL for IPN. We recommend using SSL.
custom	No	text	1-1000	Custom value pased back to the merchant IPN.

IPN Response

All responses from the IPN system will be send out as POST data. In PHP you will need to pick this up by \$_POST, and not \$_GET.

Variable	Type	Length	Description
item_name	text	1-255	"item_name" provided by merchant.
item_number	int	0-11	"item_number" provided by merchant.
txn_id	text	16-17	LitePay transaction ID.
mc_gross	int	1-11	Amount paid for by user.
receiver_email	text	1-255	"business" value provided by merchant.
payer_email	text	1-255	Buyer's inputed email address.
invoice	text	1-255	"invoice" provided by merchant.
custom	text	0-1000	"custom" provided by merchant.
payment_status	text	NA	The payment status as assigned by the server. Status can be: Completed, Refunded

Implimentation

All of the examples in this section should allow you to copy and paste them to use for yourself. If you are having trouble doing so, please visit our website and download the code samples. Remeber to always use <https://www.litepay.com> and not <http://www.litepay.com> when contacting any of LitePay sites, or many features will not function.

If you currently use a PayP*I compatable shopping cart or system, switching the payment url to <https://www.litepay.com/pay> and using only USD will most likly work with no modifications.

Request front-end

Recieving money through LitePay can be as easy as a simple link. A basic link for a user to pay would look like:

```
https://www.litepay.com/pay?business=YOUR@BUSINESS.COM -or-
https://www.litepay.com/pay?business=YOUR@BUSINESS.COM&amount=5.00&item_name=Widget
```

This may be an easy way, but most people prefer to use forms as they are easier to pass more data with. Here is an example of using a form to pass the data. Note that the custom field is a serialized array. Passing an array like this is great to pass more data than the system currently supports.

```
<form method="post" action="https://www.litepay.com/pay">
<input type="hidden" name="business" value="YOUR@BUSINESS.COM">
<input type="hidden" name="bn" value="Widget Factory">
<input type="hidden" name="item_name" value="Widget">
<input type="hidden" name="item_number" value="1">
<input type="hidden" name="return" value="http://WWW.YOURURL.COM/thank_you.php">
<input type="hidden" name="notify_url" value="http://WWW.YOURURL.COM/ipn.php">
<input type="hidden" name="invoice" value="widget-99">
<input type="hidden" name="custom" value="a:2:{s:8:"username";s:4:"hank";s:4:"date";s:4:"2006";}">
<input type="hidden" name="amount" value="10.00">
<input type="submit" value=" Pay for this fine product! ">
</form>
```

Request back-end

The request back-end primarily is a method of tagging the user before they leave your website. For example, if you have a system where products can have different proces depending on the promotion code, you would want to create an invoice in the database for the product and price. It could look like:

```
<?php

// Table 'invoice' should have a field of id set to auto_increment
$security_key = 'RANDOM ENCRYPTED KEY';
mysql_query('INSERT INTO invoice (date,product,price,key)
VALUES(NOW(),".$product.", ".$price.", ".$security_key."');
$invoice = mysql_insert_id();
```

```
?>
<form method="post" action="https://www.litepay.com/pay">
<input type="hidden" name="business" value="YOUR@BUSINESS.COM">
<input type="hidden" name="item_name" value="Widget">
<input type="hidden" name="return" value="http://WWW.YOURURL.COM/thank_you.php">
<input type="hidden" name="notify_url" value="http://WWW.YOURURL.COM/ipn.php">
<input type="hidden" name="invoice" value="<?=$invoice?>">
<input type="hidden" name="custom" value="<?=$security_key?>">
<input type="hidden" name="amount" value="10.00">
<input type="submit" value=" Pay for this fine product! ">
</form>
```

IPN Response

IPN (instant payment notification) is a must for almost all internet merchants. IPN allows your website to have instant results on the purchased goods, allowing you to deliver, or even email them right away.

For this example, assume you used the example in the "Request back-end" section. We will check that the invoice and price match what we logged in the database.

The default API callback URL. If this value is set, all transactions and subscription that do not have a callback url assigned at the time of checkout will use this one. To disable IPN at the time of checkout, set the notify_url to "none". If you change the default callback url, any transaction that is currently using the default will use the new default. You can manually edit each transaction to use a different callback url from the transaction details page. We strongly recommend using SSL to secure your callback URL. Upon entering your url it will be tested with our server to ensure it is operational. You may filter this out by looking for the string "LitePay Verify" in the user-agent.

Please keep in mind that this is a code example. If you do not understand the code directly by this example please download the full code sample archive.

```
<?php
$invoice = $_POST['invoice'];
$amount = $_POST['mc_gross'];
$user = $_POST['payer_email'];

$result = mysql_query('SELECT * FROM invoice WHERE id="'. $invoice. "'");
$row = mysql_fetch_row($result);

if (!$row['id']) {
//nothing to do
exit;

} elseif ($amount != $row['price']) {
// not enough money paid for invoice
mail($user,'Transaction Error','You have not paid in full');

} else {
// success!!
mail($user,'Transaction Success','Thank you!');
}

?>
```

