



E-Billing Solutions Pvt Ltd

This manual has been prepared for reference of E-Billings Solutions Pvt Ltd hereafter referred as EBS. This manual is intended to be used by the EBS and its employees only to facilitate use of Payment Gateway. This manual has been prepared on a best effort basis. Despite every effort to avoid errors it is likely that some errors may creep in. The EBS may at its sole discretion modify, alter, delete, add or amend any of the contents of this manual without assigning any reason or prior intimation to the Customer. It is notified that the EBS shall not be responsible for any damage or loss of action to anyone, of any kind in any manner arising by the use of this manual.

The contents of this manual are proprietary to the EBS and contain information that is privileged and confidential. Use or dissemination of the contents of this manual in any manner whatsoever is strictly prohibited. No part of this manual may be reproduced or copied in any form without the prior permission of the EBS. Breach of this condition shall be liable for legal action.]

Contents

Sno	Contents	page
1	Product Description	4
	I. Authorized Transactions	
	II. Flagged Transactions	
	III. Capturing	
	IV. Refund	
	V. Charge Back	
2	Integration Setup	4
3	Steps for Integration	5
	I. Merchant's Inputs to EBS	
	II. Return URL or Output from EBS	
4	Process Flow	8
	I. Customer's Payment Process	
	II. Merchant's Payment Process	

1. Product Description

We offer a revolutionary combination of an advanced Internet Payment Gateway with payments processed to Merchant's account with Bank. This product is called as EBS. The features of this product is

Transactions

- I. **Authorized Transactions** – This is where the customer's transactions details will land after they check out from merchant's website. The payment can be made only for the transaction in authorized status.
- II. **Flagged Transactions** – This transaction is considered to be insecure. Transactions can be approved **Authorized transactions** only after verification made by EBS team.
- III. **Capturing** – Capture transaction is that made by merchant from the pending status. This transaction is then processed by the bank which is called as **processing transaction**. Here partial capture of transaction is also possible.
- IV. **Refund** – Refund for particular transaction can be made. The flow is from Merchant to customer.
- V. **Charge Back** – The charge back is the refund made by EBS to Merchant.

2. Integration Setup

EBS Integration Kit Consists of:

a. Pay page : Pay file (pay.html) is a sample file, from where you can get the necessary information(Billing & Shipping etc..) from your customer/cardholders. Also , the page can customize at your end for your convenience based on your website requirement & platform.

b. Response Page : After the transaction, response page is used to retrieve all the DR parameter values. Return_url param value points to the response file. The value for this parameter should be in the following format,

Format: <http://www.yourdomainname.com/response.extension?DR={DR}>

C. RC4 : The DR values will be encrypted & decrypted through this file. It displays the decrypted values in the response page.

3. Steps for Integration

I. Merchant Inputs

The following will be the URLs for the external entries say a shopping chart.

From Merchants HTML page to EBS Server. This action will happen when the customer, through the merchant shopping cart and while capturing tries to establish his identity with EBS. Say:

(<https://secure.ebs.in/pg/ma/sale/pay/>) this URL is using POST method.

Parameter	Description	Type	Min	Max	-
account_id	Your Account ID	-	-	-	Required
reference_no	Your Reference Number	Char	1	20	Required
amount	Total Sale Amount	Decimal	1	14,2	Required
mode	Mode of the LIVE => live, TEST => test	Char	LIVE or TEST	LIVE or TEST	Required
description	Detail description of the sale	Char	1	255	Required
return_url	This is the url you want EBS to return back after transaction is successful.	Char	1	255	Required
name	Customer billing Name.	Char	1	128	Required
address	Customer billing address.	Char	1	255	Required
city	Customer billing city.	Char	1	32	Required
state	Customer billing state.	Char	1	32	Required
country	Customer billing country.[3 Digit country code]	Char	3	3	Required
postal_code	Customer billing postal code.	Char	1	10	Required
phone	Customer billing phone.	Char	5	20	Required
email	Customer billing email.	Char	1	100	Required
ship_name	Customer delivery Name.	Char	1	255	Required
ship_address	Customer delivery address.	Char	1	255	Required
ship_city	Customer delivery city.	Char	1	32	Required
ship_state	Customer delivery state.	Char	1	32	Required
ship_country	Customer delivery country.[3 Digit country code]	Char	3	3	Required
ship_postal_code	Customer delivery postal code.	Char	1	10	Required
ship_phone	Customer delivery phone.	Char	5	20	Required

II . Secure Hash Validation

To validate the payment request using Secure Hash, Merchant need to Enable the Secure Hash Validation option in the merchant Backend under Account->Settings: Request Preference.

Note: If this option is ENABLED, Merchant needs to generate Secure Hash and pass this value along with other payment request Parameters Name for Secure Hash is "secure_hash"

SecureHash Generation :

secure_hash=MD5secret_key|account_id|amount|reference_no|return_url|mode

Example codes to generate secure hash :

PHP :

```
$string = "$secretKey|$account_id|$amount|$reference_no|$return_url|$mode";  
$secure_hash = md5($string);
```

Java :

```
String pass = secret_key + "|" + account_id + "|" + amount + "|" + reference_no + "|" +  
+ return_url + "|" + mode;  
MessageDigest m = MessageDigest.getInstance("MD5");  
byte[] data = pass.getBytes();  
m.update(data,0,data.length);  
BigInteger i = new BigInteger(1,m.digest());  
String secure_hash = String.format("%032X", i);
```

C# :

```
string input = secret_key + "|" + account_id + "|" + amount + "|" + reference_no + "|" +  
+ return_url + "|" + mode;  
MD5 md5 = System.Security.Cryptography.MD5.Create();  
byte[] inputBytes = System.Text.Encoding.ASCII.GetBytes (input);  
byte[] hashBytes = md5.ComputeHash (inputBytes);  
StringBuilder sb = new StringBuilder(); E-Billing Solutions Pvt Ltd Private Page 2  
for (int i = 0; i < hashBytes.Length; i++)  
{  
sb.Append (hashBytes[i].ToString ("X2"));  
}  
string secure_hash = sb.ToString();
```

To get more clarification, verify the manual through the link given below,

https://support.ebs.in/app/index.php?/default_import/Knowledgebase/Article/View/357/8/hash-based-integration

III. Return URL or Output

In this sample response page, you have to edit the secret key(DEMO Key :ebskey) with your registered secret key. To get your secret Key,

- Lo-gin to the EBS Merchant Account panel through "secure.ebs.in"
- At left side of the menu, Click on the tab Account -> "Settings" to view the account details (Account ID & Secret key).

Note: To reset the secret key merchant should send an Email to "support@ebs.in" from the registered Mail id of EBS.

The response values will be retrieved through return URL along with the DR parameter values listed below. The response given back using GET method.

The Return URL has to be in the following format to receive the response from EBS Server,

<http://www.yourdomainname.com/response.extension?DR={DR}>

Note: ".extension" represents the extension based on platform you are using.

S.No	Parameter	Description
1	ResponseCode	Whether the payment is successful or not
2	ResponseMessage	Appropriate message explaining about successful or un successful payment
3	DateCreated	payment happened date
4	PaymentID	For that particular payment a id will be created for our reference
5	MerchantRefNo	Merchants reference number
6	Amount	Payment Amount
7	Mode	TEST or LIVE
8	BillingName	Customer billing Name
9	BillingAddress	Customer billing address.
10	BillingCity	Customer billing city.
11	BillingState	Customer billing state.
12	BillingPostalCode	Customer billing postal code.
13	BillingCountry	Customer billing country.[
14	BillingPhone	Customer billing phone.
15	BillingEmail	Customer billing email.
16	DeliveryName	Customer delivery Name
17	DeliveryAddress	Customer delivery address.

18	DeliveryCity	Customer delivery city.
19	DeliveryState	Customer delivery state
20	DeliveryPostalCode	Customer delivery postal code.
21	DeliveryCountry	Customer delivery country.
22	DeliveryPhone	Customer delivery phone.
23	TransactionID	Transaction ID corresponding to the transaction done
24	IsFlagged	Can know whether Transaction is Flagged or Not

Note:

Check the **ResponseCode** for the transaction from the EBS. This is the most important result field and indicates the status of the transaction. If the **ResponseCode** doesn't exist, this is an error occurred during the transaction. The **ResponseMessage** is appropriate description for the **ResponseCode**.

If the **ResponseCode** is '0' the transaction is successful.

If the **ResponseCode** is not '0' the transaction has been declined or an error condition as detected, such as the customer typed an invalid card number.

Recommended validation in the response file to avoid duplication of records in your backend

- 1) Pass your order id in the reference_no parameter and in response file, check whether MerchantRefNo and your order id matches.
- 2) Check the amount of the order and the "amount" parameter in the response is equal

If both the conditions results as true, insert order in your back end, which nullifies the duplications.

Is Flagged :

Flagged transactions are potentially risky transactions, which will be verified by our Risk team and updated manually as authorized/canceled within 24 hours.

If required, we may contact you to provide details/confirmation for the action to be taken for such risky payments.

Note : Please do not process/ship the product/service to the customer until the transaction is approved/authorized and captured.

IsFlagged can have two boolean values YES/NO

"Yes", indicates that the transaction is waiting for manual verification.

"No", indicates that it is authorized and ready for capture.

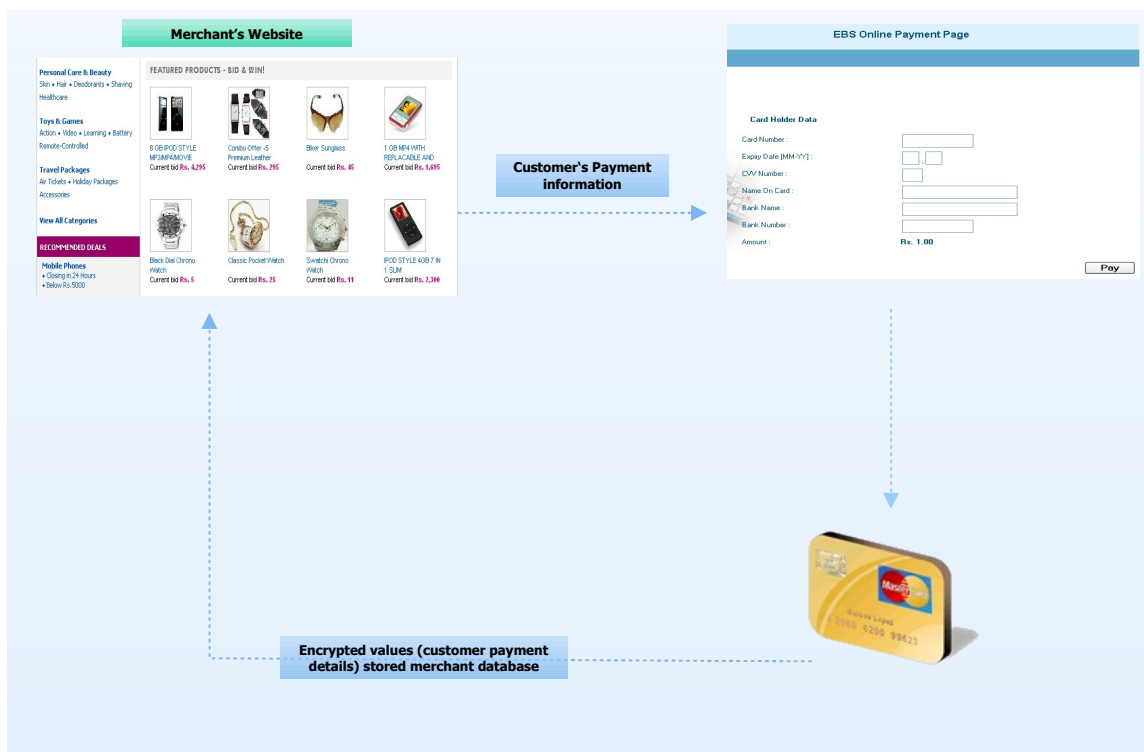
If the transaction is successful(response code= "0") and if the IsFlagged is "No", then the service can be provided immediately.

If the transaction is successful(response code = "0") and if the IsFlagged is "Yes", then the service for the corresponding payment has to be kept on hold until the manual verification is completed and authorized by EBS.

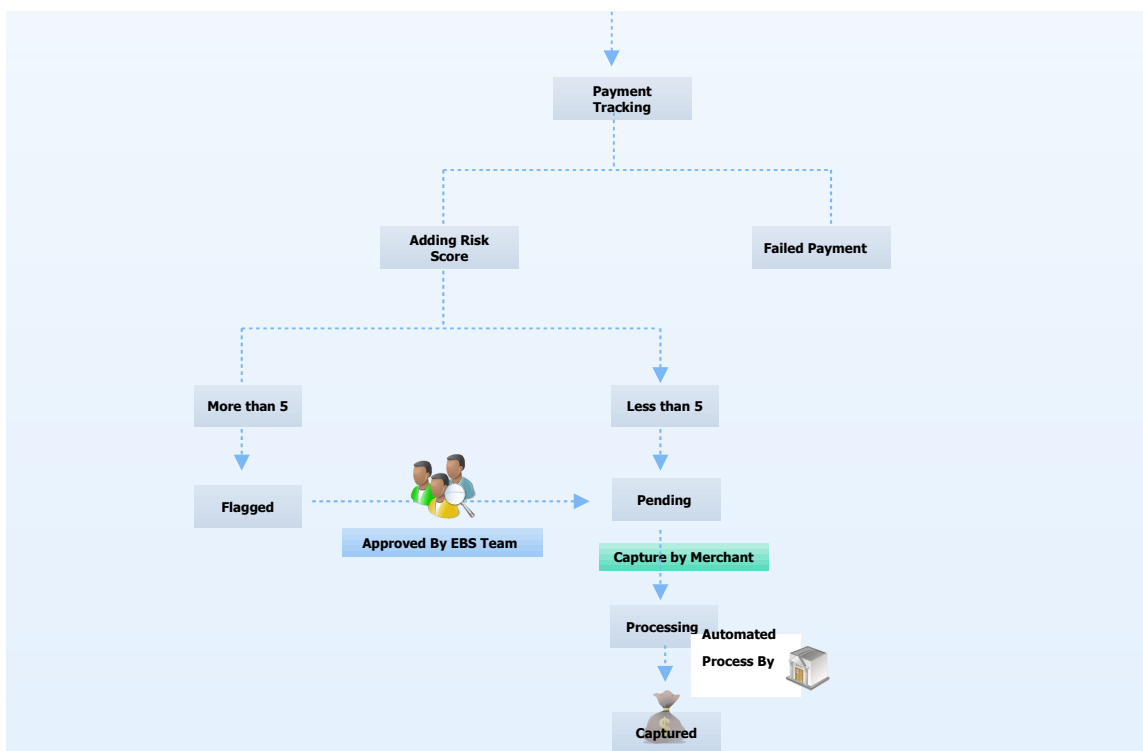
Session Time Out: - 15 minutes

5. Process Flow

I. Customer's Payment Process



II. Merchant's Payment Process





To ensure prompt support, EBS has a simple online ticketing system accessible 24x7 at <http://support.ebs.in>. Each ticket will be responded in 8 to 24 Business working Hours, with ticket number to facilitate status check anytime thereafter.

Please send your suggestions/ feedback to:

support@ebs.in

[+91 44 4231 3333](tel:+914442313333)