

FlexPay Purchase API v3

Author:	Verotel IT Services
Related documents:	FlexPay Subscription v3 API

Document History

No.	Description	Changed by:	Date:
1.0	Document created	Verotel IT Services	14/4/2015
1.0.1	Added information for CardBilling	Verotel IT Services	11/6/2015
1.1	Added DDEU payment method	Verotel IT Services	15/10/2015

Table of Contents

Introduction	2
Setup	2
FlexPay Purchase interactions	4
A new purchase order starts with a "startorder" request	4
Once the initial transaction is successfully processed the enduser is redirected to the "success URL"	4
Successful initial sale postback ("OK data") is sent in the background	5
Sale status request.....	5
Technical details.....	6
Constructing purchase "startorder" request	6
Data posted to success URL upon the FlexPay Subscription transaction (OK data)	8
'OK data' postback call	8
FlexPay status page request	9
Calculating signature.....	12

Introduction

FlexPay API allows dynamically set parameters for Verotel Order Pages. This is especially useful when integrating with eCommerce or other systems which use dynamic pricing.

FlexPay version 3 offers two types of orders:

- **purchase**
- **subscription**

The main difference between purchase and subscription is that **subscription** has some duration - a time period - for which it is active while **purchase** is just a simple one-off sale.

This document provides the description of **v3 FlexPay Purchase API**.

Setup

- Create a FlexPay website in *Control Center > Setup Websites*. From the *Website detail* page you will now have access to several configuration options.

ADD A NEW WEBSITE

How do you want integrate Verotel into your website? Please, select one of the following options:

Membership - Ideal for websites with paid members' area. We take care not only of charging your customers but we will also manage their access to your members' area using Verotel Remote User Management.	>
FlexPay - Ideal for cam sites, websites with pre-paid credit or websites selling tangible products. FlexPay is really easy and simple to implement. It is perfect for one-time payments. You simply send your customer to our orderpage telling us how much we should charge him. When we successfully charge the user, your system is notified via Postback Url so you can respond to it (add credit to user, allow file download etc.)	>

- Configure the "Postback script" and the "Success page" URLs in *FlexPay options* page.

FLEXPAY OPTIONS #64233

Payment methods

Credit card

Available worldwide

Direct Debit EU

Available only in DE, AT, CH, BE, IT, NL, ES and FR

Bitcoin

Available worldwide

Postback script URL

Success URL

Save

- Select which payment methods are enabled for your purchases. Currently Credit card and Bitcoin payments are supported. At least one payment method has to be enabled.

FLEXPAY OPTIONS #64233

Payment methods

Credit card

Available worldwide

Direct Debit EU

Available only in DE, AT, CH, BE, IT, NL, ES and FR

Bitcoin

Available worldwide

- Get the FlexPay "**Signature Key**" - this can be found on the same page. This key is used for calculating signatures to ensure integrity of data in requests and postbacks ([SEE CALCULATING SIGNATURE](#)).

FLEXPAY REQUEST DATA

Shop ID and Signature Key are required to generate FlexPay signature.

Signature Key: DKeweGGsPAhc3bfJJqhbGkKEgz46GQ

- Customize the order page to match your website design

ORDERPAGE DESIGN SETUP

ORDERPAGE STYLE

Background color

Please enter a hexadecimal RGB value which is 6 characters long.

Foreground color

Please enter a hexadecimal RGB value which is 6 characters long.

ORDERPAGE LOGO

There is currently no logo for this website.

No changes

Upload new logo

[Continue](#)

- Additionally you can require the buyers to enter their billing address by enabling this option in the *Orderpage options* section. You can also restrict sales to one of the predefined territories.

ORDERPAGE OPTIONS

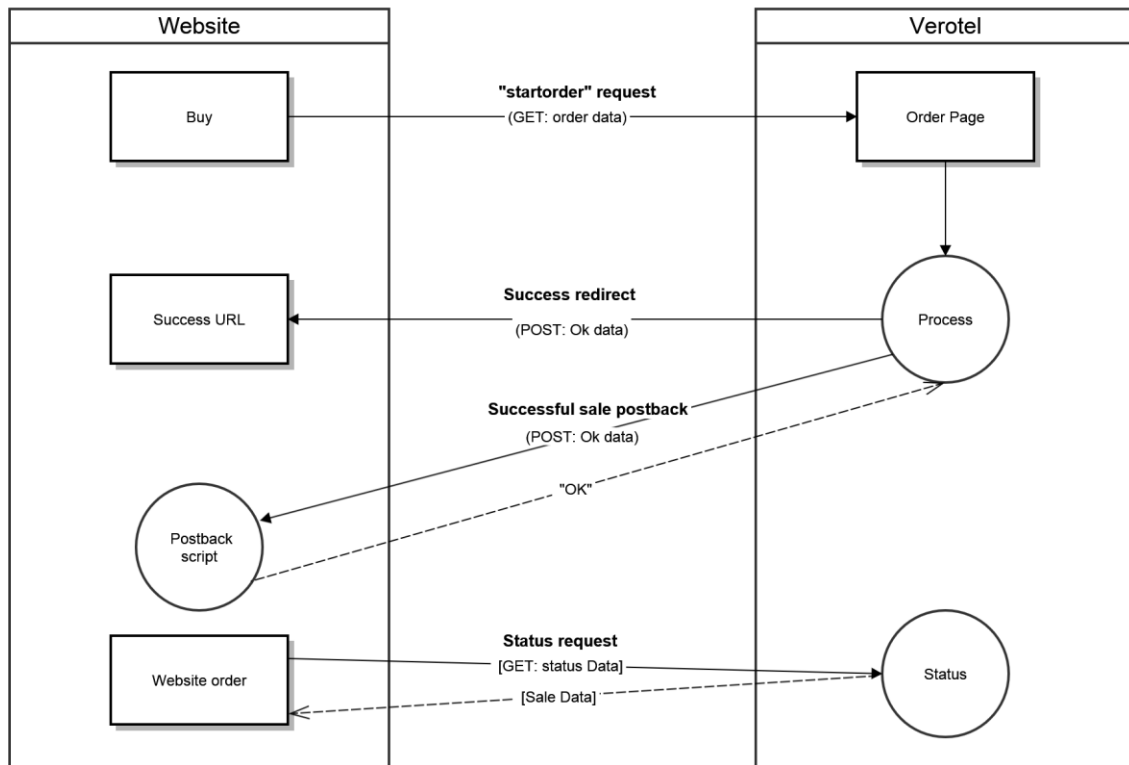
Billing address
If checked, users have to fill in the address during the order.

Require billing address

Sales territory

- NEXT: Integrate the Verotel OrderPage to your website.

FlexPay Purchase interactions



FlexPay purchase is a once-off operation without any active time period. It is an ideal dynamic pricing solution for:

- camsites pay
- per view sites
- pay per download sites
- shopping cart systems (also for tangible products)
- token / credit purchase systems

FlexPay purchase supports payments by Credit Card and Bitcoin.

A new purchase order starts with a "startorder" request

The "startorder" purchase request is used to redirect buyer to the Verotel Order Page in order to process one-off purchase with a given amount, currency and other parameters .

[SEE "CONSTRUCTING PURCHASE "STARTORDER" REQUEST"](#)

Once the initial transaction is successfully processed the enduser is redirected to the "success URL"

Verotel system also sends a set of data ("OK data") describing the sale along as HTTP parameters.

[SEE "DATA POSTED TO SUCCESS URL UPON THE FLEXPAY SUBSCRIPTION TRANSACTION \(OK DATA\)"](#)

Successful initial sale postback ("OK data") is sent in the background

The "initial" OK data postback is sent to the nominated "Postback URL" immediately after the sale has been processed.

"OK data" is sent only for successfully approved transactions. The data in the postback provide essential information about the sale. If more information is needed, for example billing address or email address of the buyer, the merchant should query the [status page](#).

Important: The Verotel system expects "OK" response. For Credit Card payments - If no such response is received, an automatic refund is initiated.

[SEE "'OK DATA' POSTBACK CALL"](#)

Sale status request

Although it is an optional step, we recommend that you cross check the received data with the Verotel Status service. You send the Verotel transaction ID or the sale Reference ID (if you have provided one in the "startorder" request) to the Verotel status script in order to verify the status of the transaction. Usually, this extra check is performed before the buyer is informed that the transaction was successfully processed. The Verotel Status service can be used anytime after the transaction took place. ([SEE "FLEXPAY STATUS PAGE REQUEST"](#))

Technical details

Constructing purchase "startorder" request

FlexPay purchase call is used to redirect buyer to the Verotel Order Page in order to process the purchase sale.

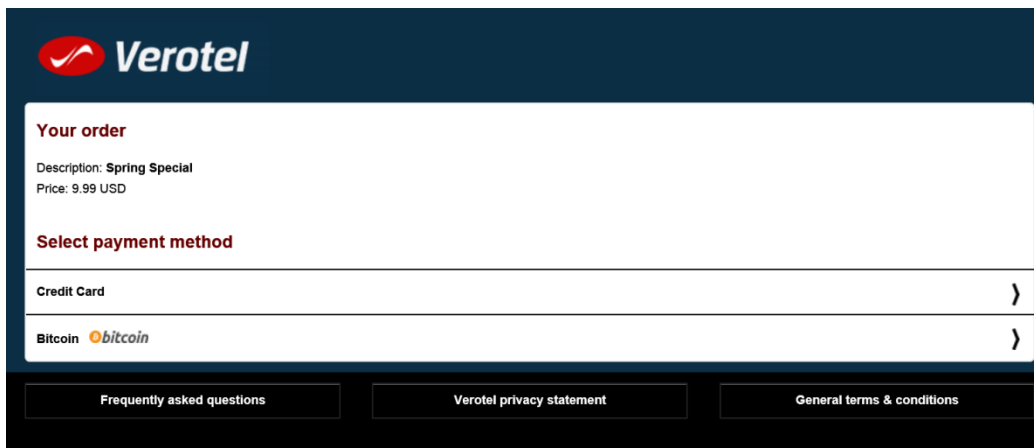
The "startorder" request for FlexPay purchase consists of number of parameters passed to "<https://secure.verotel.com/startorder?>" for Verotel accounts or "<https://secure.billing.creditcard/startorder?>" for CardBilling accounts and is secured by a signature.

Parameter	Type	Optional / Mandatory	Description
version	<i>number</i>	mandatory	version of the FlexPay call, "3" for this version
shopID	<i>number</i>	mandatory	numerical ID of the shop or website in the Verotel system
type	<i>string</i>	mandatory	"purchase"
priceAmount	<i>number</i>	mandatory	amount to be processed in nnn.nn format
priceCurrency	<i>string</i>	mandatory	3 char ISO code, must be one of the Sale currencies (USD, EUR, GBP, AUD, CAD, CHF, DKK, NOK, SEK) Note: only EUR is can be used for DDEU payment method
description	<i>string</i>	mandatory	description of the product. Text is displayed on the order page
referenceID	<i>string</i>	optional	Merchant's reference identifier. It must be unique if provided
custom1	<i>string</i>	optional	pass-through variable - max 255 printable characters
custom2	<i>string</i>	optional	pass-through variable - max 255 printable characters
custom3	<i>string</i>	optional	pass-through variable - max 255 printable characters
email	<i>string</i>	optional	email of the buyer. If not set, it will be collected on the Order Page NOTE: email is excluded from signature calculations
paymentMethod	<i>string</i>	optional	payment method, "CC", "DDEU" or "BTC" (if not set then buyers can choose from available payment methods) Note: DDEU is available only in DE, AT, CH, BE, IT, NL, ES and FR
signature	<i>string</i>	mandatory	security token to verify the integrity of the postback data (SEE CALCULATING SIGNATURE)

Example "startorder" request:

<https://secure.verotel.com/startorder?custom1=my+custom+code&description=Spring+Special&priceAmount=9.99&priceCurrency=USD&shopID=64233&type=purchase&version=3&signature=b690ae8daca52243c85d3ce4365f137944e58d1d>

Since the Payment method was not specified this opens a page which lets the buyer to select from the available payment methods. (DDEU is not offered as the sale currency is USD)



Data posted to success URL upon the FlexPay Subscription transaction (OK data)

After a successful transaction, the buyer is redirected to a nominated "Success URL" and a set of data describing the sale is sent along as HTTP parameters.

Parameter	Type	Description
shopID	<i>number</i>	numerical ID of the shop or website in the Verotel system
type	<i>string</i>	" purchase "
referenceID	<i>string</i>	Merchant's reference identifier if provided.
saleID	<i>number</i>	identifier of the sale in the Verotel system
priceAmount	<i>number</i>	amount to be processed in nnn.nn format
priceCurrency	<i>string</i>	3 char ISO code of the Sale currency
custom1	<i>string</i>	pass-through variable - max 255 printable characters
custom2	<i>string</i>	pass-through variable - max 255 printable characters
custom3	<i>string</i>	pass-through variable - max 255 printable characters
paymentMethod	<i>string</i>	Used payment method, "CC", "DDEU" or "BTC"
signature	<i>string</i>	security token to verify the integrity of the postback data (SEE CALCULATING SIGNATURE)

'OK data' postback call

The successful sale postback is sent to the nominated "**Postback URL**" immediately after the sale has been processed.

The postback is sent only for successfully approved transactions. The data in the postback provide essential information about the sale. If more information is needed, for example billing address or email address of the buyer, the merchant should query the status page.

Important: The Verotel system expects "**OK**" response. If no such response is received, an automatic refund is initiated. (Or in case of BTC payment a notification is sent to merchant, support and

enduser).

Parameter	Type	Description
shopID	<i>number</i>	numerical ID of the shop or website in the Verotel system
type	<i>string</i>	"purchase"
referenceID	<i>string</i>	merchant reference identifier. It must be unique if provided
saleID	<i>number</i>	identifier of the sale in the Verotel system
priceAmount	<i>number</i>	amount to be processed in nnn.nn format
priceCurrency	<i>string</i>	3 char ISO code of the Sale currency
custom1	<i>string</i>	pass-through variable - max 255 printable characters
custom2	<i>string</i>	pass-through variable - max 255 printable characters
custom3	<i>string</i>	pass-through variable - max 255 printable characters
paymentMethod	<i>string</i>	payment method, "CC", "DDEU" or "BTC"
signature	<i>string</i>	security token to verify the integrity of the postback data (SEE CALCULATING SIGNATURE)

FlexPay status page request

A status of a sale made with a FlexPay API can be reviewed by querying the status page. Status page provides complete information about the sale, the buyer, and its status.

Status request URL: "[https:// secure.verotel.com/status/order?](https://secure.verotel.com/status/order?)" (*Verotel*) or "<https://secure.billing.creditcard/status/order?>" (*CardBilling*) followed by parameters in the table below:

GET parameters in response	Description
version	Version of the Verotel Purchase OrderPage protocol. Currently "1", "2" or "3"
shopID	numerical ID of the website or shop in Verotel system
referenceID (optional)	Merchant's reference identifier if provided (referenceID OR saleID must be posted - NOT BOTH)
saleID (optional)	Verotel saleID identifier (referenceID OR saleID must be posted - NOT BOTH)
signature	SHA-1 hash generated on data listed above and Merchant's private key stringsha1_hex(signatureKey + ":referenceID=" + referenceID + ":saleID=" + saleID + ":shopID=" + shopID + ":version=" + version)

Example:

<https://secure.verotel.com/status/order?saleID=7263519&shopID=64233&version=3&signature=cde1607c7746ed63d6d8ec54875ed43b07895f7>

Flexpay Purchase Status response:

The date returned to the status request is in plain-text and contains lines with parameterName, colon, parameterValue:

Output parameter name	Description
response	FOUND - purchase record found and returned NOTFOUND - purchase not found ERROR - error (see 'error' key)
error	message (for response=ERROR)
saleID	identifier of the transaction in Verotel System
shopID	ID of the website or shop in Verotel system
paymentMethod	an identifier of payment method that was used for the transaction. Can be one of following: " Credit Card ", " Direct Debit EU " or " Bitcoin "
priceAmount	amount to be processed. in nnn.nn formatt
priceCurrency	3 char ISO code of the Sale currency
description	Product description text
referenceID	Merchant reference identifier
name	name of the buyer
email	email address of the buyer
country	selected or detected country ISO code (ISO 3166-1-alpha-2 code)
btc_transaction_status	Only for Bitcoin sales: The status of Bitcoin transaction as set by the postback from Bitcoin processor: " paid " - the transaction was successfully sent to the processor " complete " - the transaction was confirmed and processed completely " invalid " - the transaction could not be processed, it was unsuccessful
createdOn	Timestamp of purchase creation
saleResult	purchase processing result (APPROVED)
billingAddr_fullName	billing address: full name field value
billingAddr_company	billing address: company field value
billingAddr_addressLine1	billing address: 1st line field value
billingAddr_addressLine2	billing address: 2nd line field value
billingAddr_city	billing address: city name

Output parameter name	Description
billingAddr_zip	billing address: zip code / postal code
billingAddr_state	billing address: US state code (ISO 3166-2)
billingAddr_country	billing address: country ISO code (ISO 3166-1-alpha-2 code)

Example response:

```

response: FOUND

shopID: 60678
paymentMethod: Credit Card
priceAmount: 51.20
priceCurrency: EUR

description: some description of product
referenceID: AX62362I3
saleID: 13029033
saleResult: APPROVED
name: John Black
email: black@example.com
country: CZ

billingAddr_fullName: John Black
billingAddr_company:
billingAddr_addressLine1: Longstreet 3782/13
billingAddr_addressLine2:
billingAddr_city: London
billingAddr_zip: 73811
billingAddr_state:
billingAddr_country: GB

```

Calculating signature

The signature used in FlexPay requests and postbacks is calculated as SHA-1 hash (hexadecimal output) from the request parameters.

The first parameter has to be your signatureKey, followed by the parameters ordered alphabetically by their names.

Optional arguments that are used (have value) must be contained in the signature calculation. Optional arguments that are not used must not be contained in the signature calculation.

The email parameter in "startorder" request is not included in the signature calculations.

It is mandatory to convert arguments values into UTF-8 before computing the signature.

e.g.

```
signature = sha1_hex( signatureKey + ":description=" + description + ":period=" + period +  
":priceAmount=" + priceAmount + ":priceCurrency=" + priceCurrency + ":referenceID=" +  
referenceID + ":shopID=" + shopID + ":subscriptionType=" + subscriptionType + ":type=" + type +  
":version=" + version )
```

Example calculation:

Parameter	Value
(signatureKey)	BddJxtUBkDgFB9kj7Zwguxde4gAqha
description	Super video download
priceAmount	9.99
priceCurrency	USD
custom1	xyyyzz
shopID	64233
type	purchase
version	3

Signature calculation using the values above:

```
signature = sha1_hex(BddJxtUBkDgFB9kj7Zwguxde4gAqha:custom1=xyyyzz:description=Super video  
download:priceAmount=9.99:priceCurrency=USD:shopID=64233:type=purchase:version=3) =>  
a043071d3db1d3bbacee04e1eaf07da0d3ab1d17
```

The FlexPay subscription request then is:

```
https://secure.verotel.com/startorder?custom1=xyyyzz&description=Super+video+download&price  
Amount=9.99&priceCurrency=USD&shopID=64233&type=purchase&version=3&signature=a043071  
d3db1d3bbacee04e1eaf07da0d3ab1d17
```