# Paymentgateway Documentation

**Version**: 1.8.0

**Document start date**: 21-12-06

## Maintenance history

| Date | By | Version | Comments |
|---|---|---|---|
| 21-12-2006 | Daniel Bielefeldt | 1.0 | Document start |
| 12-09-2007 | Pelle Smidt | 1.1 | |
| 07-03-2008 | Daniel Bielefeldt | 1.2 | Added documentation for Mobile payment and better binary proxy explanation. |
| 26-06-2008 | Daniel Bielefeldt | 1.3 | Fixed uniqueorderid definition |
| 08-07-2008 | Daniel Bielefeldt | 1.4 | NEW enhanced security policy published PBS regarding merchants. APPENDIX B. Added documentation for secure proxy and payment window. |
| 29-07-2008 | Daniel Bielefeldt | 1.5 | Introduction added Reconstructed index |
| 23-04-2009 | Daniel Bielefeldt | 1.5.5 | Securitytext option is removed in section 3.1<br><br>Extra CSS class is added to submit button in secureproxy<br><br>Added new section about client requirements. |
| 04-06-2009 | Daniel Bielefeldt | 1.6.0 | Split payment feature is added to secureproxy, pamentwindow and postform api. |
| 16-07-2009 | Daniel Bielefeldt | 1.6.1 | Removed old API description. |
| 25-02-2010 | Daniel Bielefeldt | 1.6.2 | An updated is added too appendix A about md5check enforced security |
| 07-09-2010 | Daniel Bielefeldt | 1.6.3 | Documentation for extended validation and prefilled cvc value is added in section 2<br><br>Q8-LIC cardprefix has been removed<br><br>New option to show the last 4 creditcard digits. |
| 07-04-2011 | Daniel Bielefeldt | 1.6.4 | Added secureproxy examples for edankort, 3dsecure and netbank |
| 11-04-2011 | Daniel Bielefeldt | 1.6.5 | Added documentation for supplementary authorize and split capture |

| Date | By | Version | Comments |
|---|---|---|---|
| 13-04-2011 | Daniel Bielefeldt | 1.6.6 | New option to enable google analytics in paymentwindows and secureproxy |
| 30-06-2011 | Daniel Bielefeldt | 1.6.7 | Supplementary authorize can now be used with all transaction types. |
| 30-11-2011 | Daniel Bielefeldt | 1.6.8 | Added documentaion about MAC function, wich is a replacement for md5checksum.<br>An correction is added to the 3dsecure documentation. |
| 12-09-2011 | Daniel Bielefeldt | 1.6.9 | Note added to MAC description. |
| 07-12-2012 | Daniel Bielefeldt | 1.7.0 | Added versioning chapter to secureproxy documentation. |
| 26-02-2013 | Daniel Bielefeldt | 1.7.1 | Added new secureproxy placeholder for submitbutton. Note added about variables returned with callbackurl. |
| 16-09-2013 | Daniel Bielefeldt | 1.7.2 | Fraud module and settings description added. |
| 10-12-2013 | Daniel Bielefeldt | 1.7.3 | Precheckresult variable does not exists in accept url.<br>Placeholder for formsubmitv2 is removed in documentation. The first placeholder for form submit does also disable after submit. |
| 11-11-2014 | Daniel Bielefeldt | 1.7.4 | Added section regarding new payment window version 2. Renamed old paymentwindow headline |
| 09-12-2014 | Daniel Bielefeldt | 1.7.5 | Field name custom_branding corrected to custombranding in section “Paymentwindow version 2” |
| 04-12-2015 | Dennis Væversted | 1.8.0 | Removed deprecated interfaces. |

# Index

# Introduction

This document describes how you implement online payment into your webshop.

**Payment window** is a quick start solution where you don’t have to make lots of adjustments. Start by setting up a simple postform, and then look at what arguments you can send to the payment window. The window will be able to display your shop logo, total price and order number. You can either choose to open the window in the same window as the shop, or make it open as a popup.

### Client requirements

1. Client browser needs to be able to handle cookies

# Hosted payment window version 2

## Payment window post arguments

### Post arguments

| Field | Arguments | Type | Description |
|---|---|---|---|
| shopid | | int(?) | A numeric id to identify the postform. |
| currency | | int(3) | Define currency for the specific transaction.<br>Use a numeric value from ISO 4217.<br>If this field is empty the gateway will use the default value. The default value is set when logging into the payment gateway webinterface. |
| amount | | int(?) | Specify an amount for this transaction.<br>You must use minor unit to specify the amount. |
| orderid | | int(19) | Orderid for the specific transaction. |
| orderidprefix | | char(4) | Orderidprefix is a way to append a prefix to the orderid. |
| paytype | creditcard<br>3dsecure<br>edankort<br>netbank | char(10) | Assign one of the following arguments to define the type of transaction. This will lock the window for one of the following payment methods. |
| authtype | auth<br>subscribe<br>suppauth | char(9) | Authtype defines how a transaction is handled. It can be made as a subscription, supplementary auth or as plain authentication. |
| checkmd5 | | char(?) | Use”checkmd5” to ensure that the postform is unchanged while the transaction is being processed. See ”appendix a” for more details. |
| uniqueorderid | true<br>false | char(5) | Use uniqueorderid to ensure that only unique orderid’s are used. |
| accepturl | | char(?) | The url where the customer is redirected to, when a successful transaction has been made. This option will overrule the accepturl defined in the webinterface. |
| declineurl | | char(?) | The url where the customer is redirected to, when a transaction has been declined. This option will overrule the declineurl defined in the webinterface. |
| callbackurl | | char(?) | Callbackurl can make a request back to a predefined URL. Use this option together with mobile payment, credicard, edankort and 3DSecure. Verify the payment by using a new get paramenter called "checkmd5callback" |
| lang | da, en | char(2) | Define which language to display. Default language is Danish. |

| | | | |
|---|---|---|---|
| split | true<br>false | char(5) | Enable split payment. This option can split one payment into 2 or more transactions. This is useful, if your shop provides split shipping, as split payment will allow you to make one transaction, for each of every product in one order. By enabling this option amount, orderid, orderidprefix and authtype will be obsolete. Be a ware that this feature is only supported with creditcard transactions. |
| transact | | POST array | This field I used together with split payment. Every split payment is defined as an array. Example is available under section 3.2 |
| cardnomask | true<br>false | char(5) | The last 4 digits will bee attached to to the accepturl and callbackurl. The variable is named “cardnomask” |
| custombranding | | char(*) | Enter brand name. This setting will brand the window according to the colors and logo, that have been specefied in backend. |
| mac | | char(32) | Message authentication control, ensures that data isn’t tampered during data transmission. See ”appendix c” for more details. |
| protocol | 1 | Int(1) | Specify procol version. In time we will add more features, and some new features may break existing settings. Specify wich version you want to use, and future updates will not concern this installation |
| directforward | true/false | char(5) | If this option is set to true, the customer will be forwarded to your own accept or decline page, instead of the default window accept and/or decline page.<br>If set to false or empty the customer will end up seeing the default decline / accept page, and after that have the options to go directly to the custom accept or decline page. |

## Payment window post example

```
<form action="https://betaling.curanet.dk/paymentwindow/"
method="post">
<input type="hidden" name="shopid" value="SHOPID">
<input type="hidden" name="currency" value="CURRENCY (DKK 208)">
<input type="hidden" name="amount" value="AMOUNT">
<input type="hidden" name="orderid" value="ORDERID">
<input type="hidden" name="uniqueorderid" value="true">
<input type="hidden" name="lang" value="da">
<input type="hidden" name="protocol" value="1">

<input type="hidden" name="accepturl" value="ACCEPTURL">
<input type="hidden" name="declineurl" value="DECLINEURL">
</form>
```

To make use of the customaztion feature, please login to the paymentgateway backend, and choose paymentidow settings under the settings menu.
Create a brand name, and you will then be able to specify colors and upload your own shop logo.

If you choose not to forward your customers directly to the shop accept/decline page, you have the option to write your own text that will be displayed on the default accept/decline page.

# Fraud module and settings

## Introduction

Fraud detection is split into a 2-step procedure. The first check is performed, when the transaction is authorized, and the result will then indicate if the transaction needs attention. We mark the transaction with three different types. The first one is “Good” then “Warning, you may need to check this”, and the third one “most likely fraud”.

If the transaction is marked with “Warning” or “most likely fraud”, you will be able to perform a background check. The check is made against an extensive database, which contains global information about suspicious credit cards, names, addresses and more. When the background check is performed, the database will return a number that indicate how likely this transaction is fraud. The number is defined in percent where 0% is good, and 100% is bad. You will never see zero percentage change, because there is always a chance that the transaction is fraud. The key is that the lower the percentage, the better odds you have, that the transaction is clean.

To increase value of background check, we have added a few more data fields, which you can send to the gateway. Fields that should contain data about the end customer. The more details we get about the transaction, the more accurate the result is.

## Extra post fields

**Paymentwindow**

```
<input type="hidden" name="fraud_shipadr" value="Danmarksvej 26">
<input type="hidden" name="fraud_shipcity" value="Skanderborg">
<input type="hidden" name="fraud_shipregion" value="Jutland">
<input type="hidden" name="fraud_shippostal" value="8660">
<input type="hidden" name="fraud_shipcountry" value="dk">
<input type="hidden" name="fraud_custemail" value="email@email.dk">
```

## Settings

In the gateway control panel under advance settings, a menu named “Fraud handling” is added. From this menu, you are able to disable 1. Level fraud check, and also email warnings, if a transaction is marked as fraud.

Fraud is by default disabled. When fraud is disabled, the gateway will not collect any fraud information posted together with the transaction. If you later decide to enable fraud handling, fraud will then only work on transactions that are performed, after fraud has been enabled.

## Fraud reject policy

Fraud will not reject any transactions based on the result from 1. or 2. Level check. It will only inform you with relevant information, and then let you decide if the transaction is valid or not.

## Fraud result

One more column is added to the transaction list, and contains two types of information. The first icon indicates if the transaction is marked, and the second is the customer’s country. In addition, if you check transaction details, you will find further details collected for fraud handling.
If the transaction is marked as fraud, a button will appear called “background check”. This button will active the background check explained in the fraud introduction. This check can only be performed once.

# API Documentation

All API connections are based on SOAP requests. We recommend using one of the functions that are made public on our websites download section.

The public examples describe how to make connection through ASP 3.x, ASP.NET and PHP.

## Action code list

| Action code | Description |
|---|---|
| 0 | Successful action |
| 1 | Transaction declined |
| 2 | Possible fraud |
| 3 | Communication error |
| 4 | Card is expired |
| 5 | PBS internal system error |
| 6 | Invalid Transaction |
| 7 | System error |
| 8 | Wrong merchant number |
| 9 | No card record |
| 10 | Card entry below low range |
| 11 | Transaction not permitted to terminal |
| 12 | Transaction not permitted to cardholder |
| 13 | Invalid card number |
| 14 | Unauthorized content in cardnum field |
| 15 | Unauthorized content in expiry month |
| 16 | Unauthorized content in expiry year |
| 17 | Unauthorized content in CVC |
| 18 | Card number is not authorized according to cardtype. |
| 19 | Not a unique orderid |
| 20 | Empty amount |
| 21 | Not a valid md5checksum |
| 22 | Netbank authorize failed |
| 23 | Netbank authorize cancelled by customer. |
| 24 | Icash payment failed |
| 25 | MAC hash is invalid |

## Request acct authenticate example

When performing an acct authenticate, we recommend using of ASPtear or file_get_contents.
It is also possible to make this request directly from a browser.

Fill in the following parameters, and request the URL. The acct authenticate, can be requested as a batchlist, or as a single request.

When sending a single request. Just fill in the transaction ID, that was delivered whit the accepturl when the request acct was made. The second option is the amount for this specific transaction. The last option is the orderid, which can be used as a reference number between the customer order and the transaction.

Batchlist=transacnum1;amount1;orderid1

When sending more than one request, just duplicate the options, and separate them with a comma.

Batchlist=transacnum1;amount1;orderid1;orderidprefix1,transacnum2;amount2;orderid2;orderidprefix2

URL: https://betaling.curanet.dk/authsubscribe.php?batchlist=transacknum1;amount1;orderid1;orderidprefix1

The response from this action will be returned as plain text.

If a successful action was made, the response will look like this.

```
APPROVED;ORIGINALTRANSNUM;NEWTRANSNUM;AMOUNT;ORDERID;ORDERIDPREFIX
```

If the action fails, the response will look like this.

```
FAILED;ORIGINALTRANSNUM;0;AMOUNT;ORDERID;ORDERIDPREFIX
```

## Accept and decline return parameters

Following GET parameters are returned together with the accepturl and callbackurl.

| Field | Parameters | | |
|---|---|---|---|
| transacknum | A unique id which is used, to identify the transaction. The id will be available at the account balance. | | |
| orderid | Orderid provided when the authentication was requested | | |
| amount | Amount returned in minor unit | | |
| currency | Currency returned as numeric format. ISO 4217 | | |
| cardtype | **Prefix** | **Cardname** | **Country** |
| | DK | Dankort | Danish |
| | V-DK | Visa Dankort | Danish |
| | VISA(DK) | Visa Electron | Danish |
| | MC(DK) | Euro/Mastercard | Danish |
| | MC | Euro/Mastercard | Foreign |
| | MSC(DK) | Maestro | Danish |
| | MSC | Maestro | Foreign |
| | DINERS(DA) | Diners club | Danish |
| | DINERS | Diners club | Foreign |
| | AMEX(DA) | American Express | Danish |
| | AMEX | American Express | Foreign |
| | VISA | Visa | Foreign |
| | EDK | eDankort | Danish |
| | JCB | JCB | Foreign |
| | FBF | Forbrugsforeningen | Danish |
| | Q8-LIC | Q8-LIC | Danish |
| | DanskeBank | Danske Bank | Danish |
| | N/A | Unknown | Unknown |
| actioncode | A numeric value which refers to the action code list | | |

Note: All custom post variables that are sent to the gateway, will be relayed as GET parameters to callbackurl.

Following GET parameters are returned together with the declineurl.

| Field | Parameters |
|---|---|
| orderid | Orderid is provided when an authentication is requested |
| actioncode | A numeric value which refer to action code list |

Extra return values from callbackurl

| Field | Parameters |
|---|---|
| checkmd5 | Use”checkmd5” to ensure that the postform is unchanged while the transaction is processed. See ”appendix a” for more details. |

## Supplementary authorize

Supplementary authorize is a way of extend transaction default life time. If your shop is selling products that are not made available or shipped within 7 days, you can extend transaction lifetime with as many days as you want, until the product is ready to be shipped or made available for the customer.

**Update: Supplementary authorize can be used for all transaction types.  Including eDankort and 3Dsecure.**

The way supplementary authorize is working, is by setting postfield “authtype” to “suppauth”. The first card authorize will be made with zero amount, and the original amount will not be reserved on the customers bank account. This check is not for guaranteeing any coverage on the credit card, but only to make sure that the card is valid.

When the product is ready to be shipped or made available for the customer, press the capture button in the webinterface or by using the API. The capture process now makes another authorize, which first of all makes sure that the card is valid, and second, that the money is garanteed. If the capture failed, it’s properly because that the card is reported stolen, not valid anymore or that there isn’t coverage for the money on the customers bank account.

To implement supplementary authorize, please look at the “authtype” option in the arguments list.

## Split capture

Sometimes it can be usefull to split a transaction in to pieces, if the product is shipped in parts. Split capture can be activated by setting “authtype” to “suppauth”. When a transaction is made as a supplementary authorize, it will be possible to specify the amount you want to capture. The transaction is first marked as captured, when the entire amount is captured or the transaction is marked canceled.

If only part of the amount is captured, and the transaction is marked as canceled, it will still be marked as captured. Only mark the transaction as canceled if not the entire amount needs to be captured.

If the amount field is zero, when using API to capture with, the entire amount is captured.

**Note: eDankort transactions lifetime is 30 days regardless if supplementary authorize is used. This means that if split captured is used with a eDankort transactions, it has to be fully captured within 30 days.**

# Appendix

## Appendix A

MD5 checksum is used to verify data, which have been posted from the webshop to the paymentgateway. Every important data filed in the postform, is included in this verify check. We highly recommend enabling this feature.

The feature has to be enabled, before the paymentgateway will react on this field. Enabling md5checksum is done by clicking the checkbox that refer to md5checksum. The checkbox is located in the paymentgateway webinterface under “Indstillinger / Settings”.

After the checkbox is marked, there are 4 dropdown menus and one key field which have to be filled out. The most appropriate way to do this is to select a different value in all of the 4 dropdown menus. In the last key field, type in a secret that is only used in this md5checksum.

The last part is to add the md5cheksum field to the post form, and fill in the right md5checksum.

If using PHP, ASP 3.x or asp.net, we provide some examples that are made public at our websites download section.

When using split payment, the fields are added together. That means if you have 2 split transactions, you have to add both amounts and orderid, if those are the md5 criteria.

**Update!!**
Callbackurl function now has a different md5checksum to ensure that it isn’t possible to fake it. By looking under settings in the paymentgateway webinterface, you will find 5 new fileds, that applies to the callbackurl md5checksum. The new row is located below the old md5checksum. The difference between these two rows is the title that defindes them as auth and callback. The upper row is the same as always, but the new row applies to the new field that is returned as "get" parameter with the callback url. The new field is named “checkmd5callback”

# Appendix C

**Short intro about MAC function**
Message authentication code is a function that will ensure postform data isn't tampered during data transmission. This function is a replacement for the old m5checksum, that didn't include all postform data. If you are using md5checksum, we highly recommend that you upgrade to this mac function instead.

The principles are the same in MAC and md5 checksum. The big difference is that all fields that are sent to the gateway must be included in the MAC hash.

**Usage**
Login to the paymentgateway interface, and enable MAC by clicking the checkbox in the MAC section under settings and press the button "Aktiver Mac". Next step is the Mackey. The key is also generated in the same place, as where you enabled MAC. Press the button "Genererer nøgle", and a new key is displayed in the field above.

To secure your form with MAC, add a new postform field named "mac". The value is an md5 hash of every field value you are posting. Concatenate the values in the same order as they are listed in your form. In the end of the string attach the Mackey, that you generated under settings, and make an md5 hash of the entire string.

Fields you don't want to include in the hash is "checkmd5", "cardnum", "eyear", "emonth" and "cvc".

This pseudo-code shows the basic procedure of how to making the md5 hash.

```
mackey =
'67ce19247bd5765002e02db2c4cc7f81a1a7da416c29b9c3bacc9ab243c7e0564b0
c5e63f6ea0fb12a71bad8a9c564e1ea53fb9c4a23eb6bffeecb2cb4f9d03d'

string  = concatenate(shopid,
                      orderid,
                      orderidprefix,
                      currency,
                      amount,
                      mackey)

md5hash = md5(string)

# Insert md5hash value into the mac field value
<input type="hidden" name="mac" value="md5 hash value">
```

Almost the same procedure goes for the callbackurl. When Mac is enabled, a new GET variable named mac is attached to the callbackurl. The value of this GET variable is an md5 hash of every GET variables that is returned to the callbackurl including the secret key. To calculate an md5 hash, you need to concatenate the values of every returned GET variables, and attach the mackey to the end of the string. Then make an md5 hash and match it against the returned mac hash. If the hashes isn't identical, the data is tampered or you did something wrong when calculating the md5 hash.

**Using MAC with paymentwindow**
When using MAC together with paymentwindow, remember to add 2 extra fields to the MAC hash. If your not using authtype in your postform, paymentwindow will as default add authtype with value “creditcard. You then need to add “creditcard” and “true” at the end of the has string, before the mac key.