Integrate with Ingenico ePayments DirectLink (server-to-server)

ingenico
ePayments

Tabella dei contenuti

8.2 Payment methods with only maintenance via DirectLink

1. Introduction

Ingenico ePayments DirectLink allows you set up a server-to-server integration with our platform. The customer remains on a page of your own that will securely send the payment data to our servers.

You can also use DirectLink for maintenance of transactions, whether they were initiated in DirectLink or in e.g. e-Commerce mode.

Using DirectLink, there is no contact between our system and the merchant's (your) customer. Your system transmits all the information required to make the payment directly to our system in an HTTPS POST request. Our system requests the financial transaction (synchronously or asynchronously) to the relevant acquirer and returns the response to your server in XML format. Your programme reads the response and resumes its processing.

You are therefore responsible for collecting and storing your customer's confidential payment details and must guarantee the confidentiality and security of these details by means of encrypted web communication and server security.

In order to store personal and card data, you need to be PCI compliant.

2. General procedures and security settings

The following general procedures and security controls are valid for all DirectLink requests: new order requests, maintenance requests and direct queries.

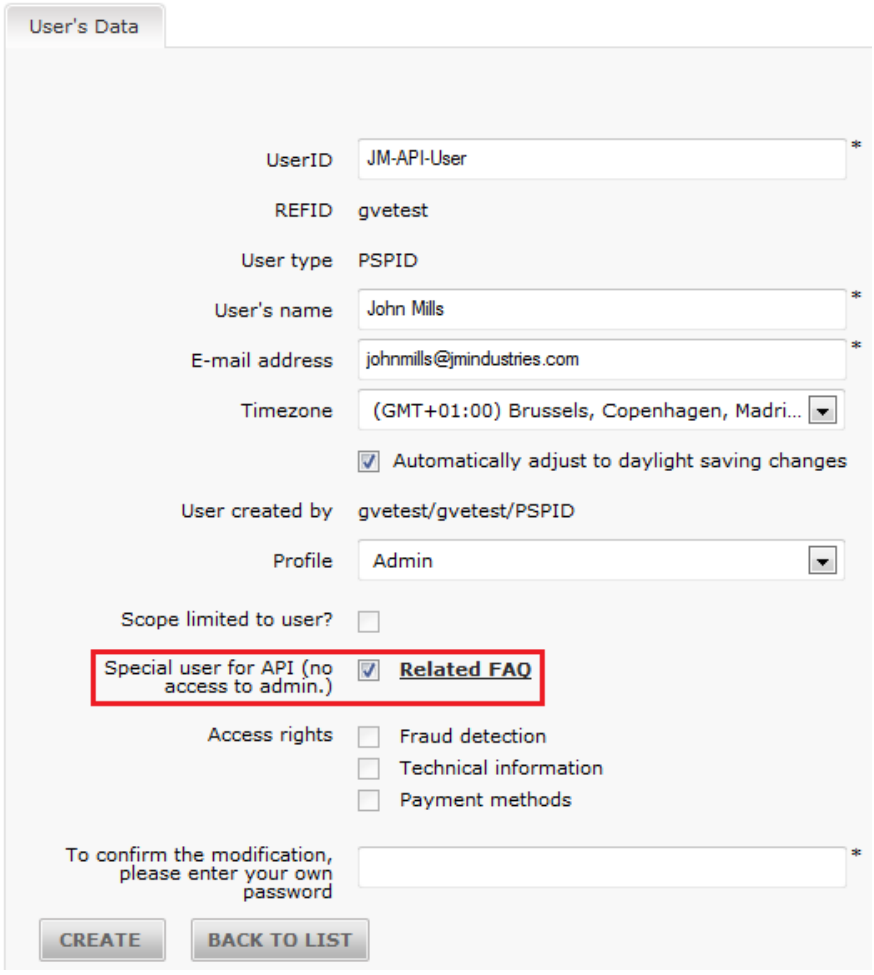
2.1 API user

An API (Application Program Interface) user is needed to make DirectLink requests with.

In general it's a user specifically designed to be used by an application to make automatic requests to the payment platform.

You can create an API user in your Ingenico ePayments account via "Configuration" > "Users". Select "New user" and fill the required fields.

To make the new user an API user, make sure to enable the "Special user for API (no access to admin.)" box.



Even though for an API user the various user profiles are available, we strongly recommend you to configure this user with the "Admin" profile.
If you want to limit the rights for maintenance of transactions (refunds, cancellations etc.), you can still change the user profile to e.g.

"Encoder".

If you are not sure, we recommend you to choose the "Admin" profile, otherwise go to User profiles (User Manager) for more information.

The password of an API user does not have to be changed regularly. This is more convenient when the password has to be hard coded into your application. However, we recommend you to change the password from time to time.

For more information about User types and how to change the API user's password, go to User types (User Manager).

2.2 Request form

For new order requests, maintenance requests and direct queries, you must send requests with certain parameters to specific URLs. The new order/maintenance/query parameters must be sent in a POST request as follows:

PSPID=value1&USERID=value2&PSWD=value3&...

The type/subtype indicating the Media Type in the Content-Type entity-header field in the POST request needs to be "application/x-www-form-urlencoded".

DirectLink works in “one request-one reply” mode; each payment is processed individually. Our system handles individual transaction requests via DirectLink and can work synchronously (where this option is technically supported), i.e. we wait for the bank’s reply before returning an XML response to the request.

2.3 Security

When we receive a request on our servers, we check the level of encryption and the IP address which the request was sent from.

2.3.1 Encryption

DirectLink is built on a robust, secure communication protocol. DirectLink API is a set of instructions submitted with standard HTTPS POST requests.

At the server end, we use a certificate delivered by Verisign. The TLS encryption guarantees that it is our servers you are communicating with and that your data is transmitted in encrypted form. There is no need for a client TLS certificate.

When we receive a request, we check the level of encryption. We allow merchants to connect to us only in secure https mode using TLS protocols and we strongly recommend to use the most recent and secure versions which are currently TLS 1.1 and 1.2.

2.3.2 IP address

For each request, our system checks the IP address from which the request originates to ensure the requests are being sent from your (the merchant’s) server. In the IP address field in the "Checks for DirectLink" section of the "Data and origin verification" tab in your account's Technical Information page, you must enter the IP address(es) or IP address range(s) of the servers that send your requests.

If the originating IP address has not been declared in the given IP address field, you will receive the error message “unknown order/1/i/”. The IP address the request was sent from will also be displayed in the error message.

2.3.3 SHA signature

The SHA signature is based on the principle of your (the merchant’s) server generating a unique character string for each order, hashed with the SHA-1, SHA-256 or SHA-512 algorithms. The result of this hash is then sent to us in your order request. Our system reconstructs this signature to check the integrity of the order data sent to us in the request.

Go to SHA-IN Signature (Ingenico ePayments e-Commerce documentation) - the principle is the same in e-Commerce and DirectLink mode.

For DirectLink, the SHA-IN passphrase needs to be configured in the "Checks for DirectLink" section of the "Data and origin verification" tab in your Technical information page.

2.4 Response parsing

We will return an XML response to your request. Please ensure that your systems parse this XML response as tolerantly as possible to avoid issues in the future, e.g. avoid case-sensitive attribute names, do not prescribe a specific order for the attributes returned in responses, ensure that new attributes in the response will not cause issues, etc.

3. Request a new order

3.1 Request URL

- The request URL in the TEST environment is https://ogone.test.v-psp.com/ncol/test/orderdirect.asp.
- The request URL in the PRODUCTION environment is https://secure.ogone.com/ncol/prod/orderdirect.asp.

Change "test" to "prod"

Replace “test” with “prod” in the request URL when you switch to your production account. If you forget to change the request URL, once you start in production with real orders, your transactions will be sent to the test environment and will not be processed by the acquirers/banks.

3.2 Request parameters

The following table contains the request parameters for sending a new order request:

Format: AN= Alphanumeric / N=Numeric, maximum allowed amount of characters

Field	Description	Format	Mandatory
PSPID	Your affiliation name in our system.	AN, 30	Yes
ORDERID	Your unique order number (merchant reference).	AN, 40	Yes
USERID	Name of your application (API) user. Please refer to the User Manager documentation for information on how to create an API user.	AN, 20 (min 2)	Yes
PSWD	Password of the API user (USERID).	AN	Yes
AMOUNT	Amount to be paid, MULTIPLIED BY 100 as the format of the amount must not contain any decimals or other separators.	N, 15	Yes
CURRENCY	ISO alpha order currency code, for example: EUR, USD, GBP, CHF, etc.	AN, 3	Yes
CARDNO	Card/account number.	AN, 21	Yes
ED	Expiry date.	MM/YY or MMYY	Yes
COM	Order description.	AN, 100	No
CN	Customer name.	AN, 35	No
EMAIL	Customer’s email address.	AN, 50	No
SHASIGN	Signature (hashed string) to authenticate the data (see SHA-IN Signature).	AN, 128	Yes
CVC	Card Verification Code. Depending on the card brand, the verification code will be a 3- or 4-digit code on the front or rear of the card, an issue	N, 5	Yes

Field	Description	Format	Mandatory
	number, a start date or a date of birth.		
ECOM_PAYMENT_ CARD_VERIFICATION	Alternative to CVC: date of birth / issue number / etc. (depending on country/bank)	N, 5	No
OWNERADDRESS	Customer's street name and number.	AN, 50	No
OWNERZIP	Customer's postcode.	AN, 10	No
OWNERTOWN	Customer's town/city name.	AN, 40	No
OWNERCTY	Customer's country, e.g. BE, NL, FR, etc.	AN, 2	No
OWNERTELNO	Customer's telephone number.	AN, 30	No
OPERATION	Defines the type of requested transaction. You can configure a default operation (payment procedure) in the "Global transaction parameters" tab, "Default operation code" section of the Technical Information page. When you send an operation value in the request, this will overwrite the default value. Possible values: • RES: request for authorisation • SAL: request for direct sale • RFD: refund, not linked to a previous payment, so not a maintenance operation on an existing transaction (you can not use this operation without specific permission from your acquirer). Optional: • PAU: Request for pre-authorisation: In agreement with your acquirer you can use this operation code to temporarily reserve funds on a customer's card. This is a common practice in the travel and rental industry. PAU/pre-authorisation can currently only be used on MasterCard transactions and is supported by selected acquirers. This operation code cannot be set as the default in your Ingenico ePayments account. Should you use PAU on transactions via acquirers or with card brands that don't support pre-authorisation, these transactions will not be blocked but processed as normal (RES) authorisations.	A, 3	Yes
WITHROOT	Adds a root element to our XML response. Possible values: 'Y' or empty.	Y or <empty>	No
REMOTE_ADDR	Customer's IP address (for Fraud Detection Module only). If a country check does not need to be performed on the IP address, send 'NONE'.	AN	No
RTIMEOUT	Request timeout for the transaction (in seconds, value between 30 and 90) Important: The value you set here must be smaller than the time out	N, 2	No

Field	Description	Format	Mandatory
	value in your system (!)		
ECI	Electronic Commerce Indicator. You can configure a default ECI value in your account's Technical information page, "Global transaction parameters" tab, "Default ECI value" section. When you send an ECI value in the request, this will override the default ECI value. Possible (numeric) values: 0 - Swiped 1 - Manually keyed (MOTO) (card not present) 2 - Recurring (from MOTO) 3 - Instalment payments 4 - Manually keyed, card present 7 - E-commerce with SSL encryption 9 - Recurring (from e-commerce)	N, 2	No
COF_INITIATOR	Credential-on-file initiator Possible values: • CIT: A transaction initiated by a cardholder • MIT: A transaction initiated by a merchant	AN	No
COF_SCHEDULE	Credential-on-files scheduled (or unscheduled) Possible values: • SCHED: A scheduled transaction • UNSCHED: An unscheduled transaction	AN	No
COF_TRANSACTION	Credential-on-file transaction Possible values: • FIRST: A scheduled transaction • SUBEQ: Subsequent series of transaction	AN	No

The list of possible parameters to send can be longer for merchants who have activated certain options/functionalities in their accounts.
Please refer to the respective option documentation for more information on extra parameters linked to the option.
The following request parameters are mandatory in new orders:

- PSPID and USERID
- PSWD
- ORDERID
- AMOUNT (x 100)
- CURRENCY
- CARDNO
- ED
- CVC

- OPERATION

3.3 Test page

Our test page to send order requests in DirectLink can be found here: https://ogone.test.v-psp.com/ncol/test/testodl.asp.

3.4 Excluding specific payment methods

If there are payment methods you don't want a customer to be able to pay with, you can use a parameter to do so.
This is particularly useful for sub-brands, when you want to accept a brand (e.g. MasterCard) but not one of its sub-brands (e.g. Maestro).

The parameter is the following:

Field	Usage
EXCLPMLIST	List of payment methods and/or credit card brands that should NOT be used. Values must be separated by a ";" (semicolon).

If a customer tries paying with a card linked to a payment method and/or (sub)brand thT you've excluded BY using the EXCLPMLIST parameter, the error message "Card number incorrect or incompatible" will be returned with the NCERRORPLUS return field.

3.5 Order request using 3-D Secure

Our system supports the usage of 3-D Secure with DirectLink.

Important

- If you wish to use 3-D Secure with DirectLink, you need to have the D3D option activated in your account.
- Some acquiring banks require the use of 3-D Secure. Please check with your acquirer if this is the case for you.

3.6 Split credit/debit cards

The functionality to split VISA and MasterCard into a debit and a credit payment method allows you to offer them to your customers as two different payment methods (e.g. VISA Debit and VISA Credit), or you can decide only to accept one of both split brands.

To use the split of credit and debit cards via DirectLink, you need to include the CREDITDEBIT parameter in the fields that you send to the orderdirect.asp page (and therefore also include in the SHA-IN calculation!).

Field	Format
CREDITDEBIT	"C": credit card "D": debit card

Related error: When the buyer selects the debit card method but next enters a credit card number, an error code will be returned: 'Wrong brand/Payment method was chosen'.

If the payment is successfully processed with the CREDITDEBIT parameter, the same parameter will also be returned in the XML response, and/or can be requested with a Direct Query. However, whereas the submitted values are C or D, the return values are "CREDIT" or "DEBIT".

You will also find these return values in transaction overview via "View transactions" and "Financial history", and in reports you may

download afterwards.

Configuration in your account

The "split" functionality can also be activated and configured per payment method, in your Ingenico ePayments account. Go to Split Credit/Debit Cards for more information.

3.7 Processing transactions with stored credentials

Credential-on-file (COF) transaction uses existing card details that are already stored by merchants to process the payment. Before initiating a credential-on-file (COF) transaction, the cardholder will first need to authorize the merchant to store the card details. Credential-on-file (COF) mostly applies to recurring payments and states whether the payment is initiated by a cardholder or merchant.

There are two types of credential-on-file (COF) transactions: cardholder-initiated transaction (CIT) or merchant-initiated transaction (MIT). Cardholder-initiated transaction (CIT) will always need to take place before initiating merchant-initiated transaction (MIT).

A cardholder-initiated transaction (CIT) is a transaction where the cardholder is involved in the transaction and personally authenticates the transaction, by means of a signature, 3D-Secure appliance, or presenting IDs.

Example of a cardholder-initiated transaction (CIT):

A cardholder buys a train ticket online and makes a payment. He/She makes the payment with his/her credit card and is being asked to authenticate and authorize the payment. At the same, the cardholder is also asked if he/she wants to save the credit card information related to this payment. If the cardholder agrees, this information can then be re-used in future transactions initiated by the merchant.

A merchant-initiated transaction (MIT) is a transaction initiated by a merchant that acts as a follow-up to a cardholder-initiated transaction (CIT) and a pre-agreed standing order for goods and services purchased by the cardholder. The cardholder does not have to be involved in the transaction.

Example of a merchant-initiated transaction (MIT):

A merchant can automatically initiate a transaction to fulfill a cardholder's payment on a monthly magazine subscription.

In compliance with the regulations set by Visa and MasterCard for credential-on-file (COF) transaction, new parameters need to be sent to determine the COF transaction.

Impacted if:

- You are using an Alias
- You plan to initiate recurring transactions (scheduled or not) after initiating a cardholder-initiated transaction (CIT) for the first time

Required action

By default, these parameters are used in a DirectLink Server-to-Server transaction:

Parameters	Description
CIT-FIRST- UNSCHEDULED	Applies when an alias is used or created

Parameters	Description
CIT-FIRST- SCHEDULED	Applies to a first scheduled payment/subscription
MIT-SUBSEQUENT-UNSCHEDULED	Applies when an alias is used or created
MIT-SUBSEQUENT-SCHEDULED	Applies to installment

The default values are flagged if you don't add any parameters. However, if you want to change it, you can overwrite these default values by sending the new parameters. Do not forget to recalculate the SHA signature as well (**click here** for more information about SHA signature).

Parameters	Values	Description
COF_INITIATOR	CIT	A transaction initiated by a cardholder
	MIT	A transaction initiated by a merchant
COF_SCHEDULE	SCHED	A scheduled transaction
	UNSCHED	An unscheduled transaction
COF_TRANSACTION	FIRST	First of a series of transactions
	SUBEQ	Subsequent series of transactions

4. Order response

Our server returns an XML response to e request:

Example of an XML response to an order request

```
<?xml version=”1.0”?>
<ncresponse orderID=”99999” PAYID=”1111111” NCSTATUS=”0” NCERROR=”” NCERRORPLUS=”” ACCEPTANCE=”12345”
STATUS=”5” ECI=”7” amount="125" currency="EUR" PM="CreditCard" BRAND="VISA"/>
```

The following table contains a list of the ncresponse tag attributes:

Field	Description
ACCEPTANCE	Acceptance code returned by acquirer.
amount	Order amount (not multiplied by 100).
BRAND	Card brand or similar information for other payment methods.
currency	Order currency.
ECI	Electronic Commerce Indicator.
NCERROR	Error code.
NCERRORPLUS	Explanation of the error code.
NCSTATUS	First digit of NCERROR.
orderID	Your order reference.
PAYID	Payment reference in our system.
PM	Payment method.
STATUS	Transaction status. (Possible statuses)

The attribute list may be longer for merchants who have activated certain options (e.g. the Fraud Detection) in their accounts. Please refer to the respective option documentation for further information about additional response attributes linked to the option.

4.1 Duplicate request

If you request processing for an already existing (and correctly processed) orderID, our XML response will contain the PAYID corresponding to the existing orderID, the ACCEPTANCE given by the acquirer in the previous processing, STATUS “0” and NCERROR “50001113”.

5. Direct Maintenance

A direct maintenance request from your application allows you to:

- Perform a data capture (payment) of an authorised order automatically (as opposed to manually in the back office);
- Cancel an authorisation of an order;
- Renew an authorisation of an order;
- Refund a paid order.

Data captures, authorisation cancellations and authorisation renewals are specifically for merchants who have configured their account/requests to perform the authorisation and the data capture in two steps.

5.1 Maintenance request

5.1.1 Request URL

- The request URL in the TEST environment is https://ogone.test.v-psp.com/ncol/test/maintenancedirect.asp.
- The request URL in the PRODUCTION environment is https://secure.ogone.com/ncol/prod/maintenancedirect.asp.

Change "test" to "prod"

Replace “test” with “prod” in the request URL when you switch to your production account. If you forget to change the request URL, once you start working with real orders, your maintenance transactions will be sent to the test environment and will not be sent to the acquirers/banks.

5.1.2 Request parameters

The following table contains the mandatory request parameters for performing a maintenance operation:

Field	Description
AMOUNT	Order amount multiplied by 100. This is only required when the amount of the maintenance differs from the amount of the original authorisation. However, we recommend its use in all cases. Our system will check that the maintenance transaction amount is not higher than the authorisation/payment amount.
OPERATION	Possible values: • REN: renewal of authorisation, if the original authorisation is no longer valid. • DEL: delete authorisation, leaving the transaction open for further potential maintenance operations. • DES: delete authorisation, closing the transaction after this operation. • SAL: partial data capture (payment), leaving the transaction open for another potential data capture. • SAS: (last) partial or full data capture (payment), closing the transaction (for further data captures) after this data capture. • RFD: partial refund (on a paid order), leaving the transaction open for another potential refund. • RFS: (last) partial or full refund (on a paid order), closing the transaction after this refund. Please note that with DEL and DES that not all acquirers support the deletion of an authorisation. If your acquirer does not support DEL/DES, we will nevertheless simulate the deletion of the authorisation in the back office.

Field	Description
ORDERID	You can send the PAYID or the orderID to identify the original order. We recommend the use of the PAYID.
PAYID	
PSPID	Your account's PSPID
PSWD	Password of your API-user
SHASIGN	Signature (hashed string) to authenticate the data (see SHA-IN-signature)
USERID	Your API-user

5.1.3 Test page

You can test direct maintenance requests here: https://ogone.test.v-psp.com/ncol/test/testdm.asp

5.2 Maintenance response

Our server returns an XML response to the maintenance request:

Example of an XML response to a direct maintenance request

```
<?xml version=”1.0”?>
<ncresponse orderID=”99999” PAYID=”1111111” PAYIDSUB=”3” NCSTATUS=”0” NCERROR=”” NCERRORPLUS=””
ACCEPTANCE=”12345” STATUS="91" amount="125" currency="EUR"/>
```

The following table contains a list of the ncresponse tag attributes:

Field	Description
ACCEPTANCE	Acceptance code returned by acquirer
AMOUNT	Order amount (not multiplied by 100)
CURRENCY	Order currency
NCERROR	Error code
NCERRORPLUS	Explanation of the error code
NCSTATUS	First digit of NCERROR
ORDERID	Your order reference
PAYID	Payment reference in our system
PAYIDSUB	The history level ID of the maintenance operation on the PAYID
STATUS	Transaction status (Possible statuses)

The standard ncresponse tag attributes are the same as those for the XML reply to a new order, except for the extra attribute PAYIDSUB.

5.3 Duplicate request

If maintenance is requested twice for the same order, the second request will theoretically be declined with an error “50001127” (This order is not authorised), because the initial successful transaction will have changed the order status.

6. Direct Query

A direct query request from your application allows you to query the status of an order automatically (as opposed to manually in the back office). You can only query one payment at a time, and you will only receive a limited amount of information about the order.

If you need more details about the order, you can look up the transaction in the back office or perform a manual or automatic file download (see Consult your transactions and Batch).

6.1 Query request

6.1.1 Request URL

- The request URL in the TEST environment is https://ogone.test.v-psp.com/ncol/test/querydirect.asp
- The request URL in the PRODUCTION environment is https://secure.ogone.com/ncol/prod/querydirect.asp

Change "test" to "prod"

Replace “test” with “prod” in the request URL when you switch to your production account.

6.1.2 Request parameters

The following table contains the mandatory request parameters to perform a direct query:

Field	Description
ORDERID	You can send the PAYID or the ORDERID to identify the original order. We recommend the use of the PAYID.
PAYID	
PAYIDSUB	You can indicate the history level ID if you use the PAYID to identify the original order (optional).
PSPID	Your account's PSPID
PSWD	Password of your API-user
USERID	Your API-user

6.1.3 Test page

You can test direct query requests here: https://ogone.test.v-psp.com/ncol/test/testdq.asp.

6.2 Query response

Our server returns an XML response to the request:

Example of an XML response to a direct query

```
<?xml version=”1.0”?>
<ncresponse orderID=”99999” PAYID=”1111111” PAYIDSUB=”3” NCSTATUS=”0” NCERROR=”” NCERRORPLUS=””
ACCEPTANCE=”12345” STATUS="9" ECI=”7” amount="125" currency="EUR" PM="CreditCard" BRAND="VISA"
CARDNO="XXXXXXXXXXXX1111" IP="212.33.102.55"/>
```

The following table contains a list of the ncresponse tag attributes:

Field	Usage
ACCEPTANCE	Acceptance code returned by acquirer
amount	Order amount (not multiplied by 100)
BRAND	Card brand or similar information for other payment methods
CARDNO	The masked credit card number
currency	Order currency
ECI	Electronic Commerce Indicator
IP	Customer's IP address, as detected by our system in a 3-tier integration, or sent to us by the merchant in a 2-tier integration
NCERROR	Error code
NCERRORPLUS	Explanation of the error code
NCSTATUS	First digit of NCERROR
orderID	Your order reference
PAYID	Payment reference in our system
PAYIDSUB	The history level ID of the maintenance operation on the PAYID
PM	Payment method
STATUS	Transaction status

The standard ncresponse tag attributes are identical to those for the XML reply to a new order, except for the additional attributes PAYIDSUB, CARDNO and IP.

The attribute list may be longer for merchants who have activated certain options (e.g. the Fraud Detection) in their accounts. Please refer to the respective option documentation for more information on extra response attributes linked to the option.

6.2.1 Transactions processed with e-Commerce (hosted payment page)

If the transaction whose status you want to check was processed with e-Commerce (hosted payment page), you may also receive the following additional attributes (providing you sent these fields with the original e-Commerce transaction).

Field	Description
complus*	A value you wanted to have returned
(paramplus content)*	The parameters and their values you wanted to have returned

*Please check the Variable feedback parameters (e-Commerce documentation).

Example of an XML response to a direct query for an e-Commerce transaction

<ncresponse orderID=”99999” PAYID=”1111111” PAYIDSUB=”3” NCSTATUS=”0” NCERROR=”” NCERRORPLUS=”” ACCEPTANCE=”12345” STATUS="9" amount="125" currency="EUR" PM="CreditCard" BRAND="VISA" CARDNO="XXXXXXXXXXXX1111" IP="212.33.102.55" COMPLUS="12345678912345678912345678 9" SessionID="126548354" ShopperID="73541312"/>

6.3 Possible response statuses

The STATUS field will contain the status of the transaction (see Possible statuses).

Only the following status is specifically related to the query itself:

Status	NCERROR	NCSTATUS	Description
88			The query on querydirect.asp failed

6.4 Direct Query as fallback

The response times for a DirectLink transaction request are generally a few seconds; however, some acquirers may have longer response times.

If you haven't received a response from our system after 30 seconds, you can send a request to querydirect.asp, asking for the status of your most recent transaction sent to orderdirect.asp. If you receive an immediate reply containing a non-final status for the transaction, there might be issues on the acquirer's end.

If you haven't received an answer to this direct query request after 10 seconds, there might be issues on our end. You can repeat this request to querydirect.asp every 30 seconds until you see you receive a response within 10 seconds.

Note

- This check system will only be able to pinpoint issues at our end if there is also a check at your end to verify that requests are leaving your servers correctly.
- An issue at our end will not always necessarily be caused by downtime, but could also be as a result of slow response times due to database issues for example.
- Please use these checks judiciously to avoid bombarding our servers with requests, otherwise we might have to restrict your access to the querydirect.asp page.

Important

To protect our system from unnecessary overloads, we prohibit system-up checks which involve sending fake transactions or systematic queries, as well as systematic queries to obtain transaction feedback for each transaction.

7. Richiesta al controllore dei dati in merito all'informativa sulla privacy

In base agli articoli 12, 13 e 14 del Regolamento GDPR, il controllore dei dati è tenuto a informare i clienti finali in merito all'elaborazione futura dei loro dati personali. Tali informazioni devono essere personalizzate in base al tipo di dati personali da inserire per una transazione specifica (ad es. metodo di pagamento selezionato, controllore/addetto all'elaborazione, acquirente, truffa). Il risultato deve essere disponibile e visibile al momento della raccolta dei dati e al titolare dei dati deve essere offerta una copia da stampare e da scaricare. Ai sensi del regolamento GDPR, è necessario mostrare l'informativa al cliente prima che convalidi la transazione. In uno scenario ideale, questi dati devono essere visualizzati sulla stessa pagina in cui il cliente inserisce le credenziali della carta/del conto.
La seguente richiesta relativa all'informativa sulla privacy permette di recuperare tutti i dati da mostrare al cliente in merito ai nostri servizi per garantire la conformità al regolamento GDPR.

7.1 Query request

7.1.1 URL della richiesta

• L'URL della richiesta nell'ambiente di prova è https://secure.ogone.com/ncol/test/privacy-policy.asp

• L'URL della richiesta nell'ambiente di PRODUZIONE è https://secure.ogone.com/ncol/prod/privacy-policy.asp

Modifica da "test" a "prod"

Quando si passa all'account di produzione, sostituire "test" con "prod" nell'URL della richiesta.

7.1.2 Parametri della richiesta

La seguente tabella contiene i parametri obbligatori della richiesta da trasmettere al cliente in merito all'uso dei suoi dati sensibili:

Campo	Formato	Descrizione
USERID	Stringa	Utente API
PSWD	Stringa	Password dell'utente API
PSPID	Stringa	PSPID dell'account
BRAND	Stringa (ad es. Visa)	Facoltativo: marca del metodo di pagamento È possibile inviare diverse volte questo campo per ottenere i risultati di diversi marchi in una volta sola. • Non trasmettere il marchio equivale a trasmettere tutti i marchi attivi. • I marchi vuoti o con il formato errato vengono ignorati.
LANGUAGE	ISO 639-1: codici con due lettere (ad es. FR)	Facoltativo: la lingua in cui si desidera recuperare il testo. Se non si specifica alcun valore, il testo verrà restituito nella lingua configurata del commerciante.

7.1.3 Pagina di prova

È possibile testare qui le richieste di query diretta: https://secure.ogone.com/ncol/test/privacy-policy.asp

7.2 Risposta alla query

Segue un elenco degli elementi XML e degli esempi di risposte XML restituite per vari risultati.

Nome	Formato	Descrizione

Risposta	Complesso	Nodo principale, sempre presente
Response.Status	Stringa, possibili valori: Success, SuccessWithWarnings, Error	Sempre presente
Response.Body	Complesso	Presente soltanto se Response.Status = Success o SuccessWithWarnings
Response.Body.Html	Stringa / html	Vuoto se Response.Status = SuccessWithWarnings e Response.Warnings.Warning.Code = NoContent
Response.Errors	Complesso	Presente soltanto se Response.Status = Error
Response.Errors.Error	Complesso	Presente varie volte in un nodo <Errors>
Response.Warnings	Complesso	Presente soltanto se Response.Status = SuccessWithWarnings oppure Error
Response.Warnings.Warning	Complesso	Presente varie volte in un nodo <Warnings>
Response.Errors.Error.Code Response.Warnings.Warning.Code	Stringa, possibili valori: •All'interno di un nodo <Error>: Unauthorized, InternalServerError •All'interno di un nodo <Warning>: NoContent	Sempre presente in un nodo <Error> oppure <Warning>
Response.Errors.Error.Message Response.Warnings.Warning.Message	Stringa	Facoltativo

Se si trova Response.Status=Error, consultare Response.Errors.Error per risolverlo.

Seguono due esempi di operazioni eseguite:
1. Esempio di risposta XML in caso di operazione eseguita con avvisi. Viene restituita se nessun dato sensibile deve essere rivelato al cliente.

```
<?xml version="1.0" encoding="utf-8"?>
 <Response>
    <Status>SuccessWithWarnings</Status>
    <Warnings>
        <Warning>
            <Code>NoContent</Code>
        </Warning>
    </Warnings>
    <Body>
        <Html/>
    </Body>
 </Response>
```

2. Esempio di risposta XML in caso di operazione eseguita con contenuto. Nell'esempio viene mostrata la sezione 2.

```
<?xml version="1.0" encoding="utf-8"?>
<Response>
    <Status>Success</Status>
    <Body>
        <Html><![CDATA[<ul><li><h2>Title 1</h2><p>Content 1</p></li><li>
<h2>Title 2 (VISA, American Express)</h2><p>Content 2</p></li></ul>]]></Html>
    </Body>
</Response>
```

8. Payment method exceptions

For certain payment methods, the parameter values differ from the standard credit card values.

8.1 Direct Debits

8.1.1 Direct Debits AT

The following table contains the specific parameter values allowing the transmission of Direct Debit AT transactions via DirectLink.

Format: AN= Alphanumeric / N=Numeric, maximum allowed amount of characters

Field	Description	Format/Value
CARDNO	Bank account number	AN, 21 Format: XXXXXXXXXXXBLZYYYYY XXXXXXXXXXX: account number, numeric, 11 digits. YYYYY: Bank code (Bankleitzahl), 5 digits.
CN	Bank account holder's name	AN, 35
ED	Expiry date	„99/99“ oder „9999“
OPERATION	Operation code (Action to be performed)	A, 3 Possible values: • RES: authorisation • SAL/SAS: debit money from the bank account • RFD/RFS: refund money (*)
OWNERADDRESS	Address of the account holder	AN, 50
OWNERTOWN	City/town of the account holder	AN, 40
OWNERZIP	Postal code of the account holder	AN, 10
PM	Payment method	AN, 25 “Direct Debits AT”

(*If the Refund option is available and active, and DTAUS Refunds is available)

8.1.2 Direct Debits DE (ELV)

The following table contains the specific parameter values allowing the transmission of ELV transactions via DirectLink. (not Wirecard/Billpay)

Format: AN= Alphanumeric / N=Numeric, maximum allowed amount of characters

Field	Description	Format/Value	Mandatory

Field	Description	Format/Value	Mandatory
CARDNO	Bank account number	IBAN: 22 alphanumeric characters OR Bank account number + BLZ. Format: XXXXXXXXXBLZYYYYYYYY XXXXXXXXX: account number, numeric, 1 to 10 digits. YYYYYYYY: Bank code (Bankleitzahl), 8 digits.	Yes
CN	Bank account holder's name	AN, 35	Yes
ED	Expiry date	„99/99“ oder „9999“	Yes
MANDATEID	Unique mandate reference. Telego: If not provided, the platform will take the ORDERID or PAYID Note: If not provided, easycash will generate a value.	Telego: AN, 35 / Charset: “A-Z a-z 0-9 space /-?:().,'+”) If not provided, the platform will take the ORDERID or PAYID Easycash: Format: AN, 27 / Charset: “A-Z a-z 0-9 space /-?:().,'+”) Note: If not provided, easycash will generate a value.	No
OPERATION	Operation code (Action to be performed)	A, 3 Possible values: • RES: authorisation • SAL/SAS: debit money from the bank account • RFD/RFS: refund money (*)	No
OWNERADDRESS	Account holder's street name and number	AN, 50	Yes
OWNERTOWN	Account holder's city/town	AN, 40	Yes
OWNERZIP	Account holder's postal code	AN, 10	Yes
PM	Payment method	AN, 25 "Direct Debits DE”	Yes

Note: These fields can be returned in the DirectLink XML-response and need to be included in the SHA-IN calculation (optionally also SHA-OUT)

(*If the Refund option is available and active, and DTAUS Refunds is available)

8.1.3 Direct Debits NL

The following table contains the specific parameter values allowing the transmission of Direct Debits NL transactions via DirectLink.

Format: AN= Alphanumeric / N=Numeric, maximum allowed amount of characters

Field	Description	Format/Value
CARDNO	Bank account number	Regular Dutch account number: max. 10 alphanumeric characters (if less, left pad with zeros). OR IBAN account number: max. 35 alphanumeric characters (SEPA)
CN	Bank account holder's name	AN, 35
ED	Expiry date	„99/99“ oder „9999“
OPERATION	Operation code (Action to be performed)	A, 3 Possible values: • SAL or SAS: debit money from the bank account • RFD or RFS: credit money to the bank account (refund)
OWNERTOWN	City of the bank account holder	AN, 40
PM	Payment method	AN, 25 “Direct Debits NL”
Only relevant for SEPA (*) transactions:		
BIC	Bank Identifier Code	AN, 11
MANDATEID	Unique mandate reference. Note: If not provided, the ORDERID will be used.	AN, 35 No spaces; cannot start or end with a forward slash "/", or contain two consecutive slashes.
SEQUENCETYPE	The Direct Debit transaction type Note: If not provided, the transactions will be considered as a “one-off” and value "OOFF" will be used.	Possible values to indicate the Direct Debit transaction type (AN, 4): • "FRST": First collection of a series of Direct Debit instructions • "RCUR": Direct Debit instructions where the debtor's authorisation is used for regular Direct Debit transactions initiated by the creditor • "FNAL": Final collection of a series of Direct Debit instructions (afterwards same MandateID can't be used anymore) • "OOFF": Direct Debit instruction where the debtor's

Field	Description	Format/Value
		authorisation is used to initiate one single Direct Debit transaction
SIGNDATE	Date mandate was signed by the buyer. Note: If not provided, the transaction date will be used.	YYYYMMDD

(*SEPA: Single Euro Payments Area)

Note: These fields can be returned in the DirectLink XML-response and need to be included in the SHA-IN (and optionally SHA-OUT) calculation.

8.2 Payment methods with only maintenance via DirectLink

For certain payment methods (excluding credit cards), you cannot send new transactions via DirectLink, but you can send certain maintenance operations via DirectLink. This is the case for PostFinance Card, PostFinance E-finance, PayPal Express Checkout and TUNZ.

When sending maintenance operations, the PM, BRAND, CARDNO and ED fields are not required, so no specific values need to be sent for these payment methods.