## **Authorize.Net Authorize and Capture Transaction**

Authorizing and Capturing a credit card transaction is one of the most involved methods included in this integration. This method will create a new Customer Payment Profile on Authorize.Net and credit card on OrderCloud.io if one does not already exist, authorize and capture a payment transaction on Authorize.Net, create a payment on OrderCloud.io tied to the authenticated user’s current Order, and also create a transaction tied to that payment.

If the transaction authorization and capture are successful, but there is an error while creating the payment or transaction on OrderCloud.io, the Authorize.Net transaction will be voided.

### Authorize and Capture Request

This method requires either CardDetails.CreditCardID (for a previously created card) or CardDetails.CardNumber and CardDetails.ExpirationDate (to create a new card).

POST https://api.ordercloud.io/v1/integrationproxy/authorizenet HTTP/1.1

Authorization: bearer insert\_access\_token\_here

Content-Type: application/json; charset=UTF-8

{

"BuyerID": "...",

“OrderID”: “...”,

“Amount”: 0.00,

“TransactionType”: “authCaptureTransaction”,

“CardDetails”: {

“CreditCardID”: “...”,

“CardholderName”: “...”,

“CardType”: “...”,

“CardNumber”: “...”,

“ExpirationDate”: “...”,

“CardCode”: “...”

}

}

### Authorize and Capture Response

HTTP/1.1 200 OK

Content-Type: application/json; charset=UTF-8

{

"ChargeStatus": "...",

“CreditCardID”: “...”,

“PaymentID”: “...”,

“TransactionID”: “...”,

“Messages”: [

{

"code": "1",

"description": "..."

}

]

}

### Error Handling

During the authorization and capture process, the Authorize.Net payment transaction will be created first, followed by the payment and transaction on OrderCloud.io. In the case that the OrderCloud.io steps fail, the transaction will be voided automatically on Authorize.Net.

Errors will return the exact response directly from the Authorize.Net or OrderCloud.io endpoint that failed. However, if any required fields are missing, a 400 error will be returned before any of the update process is executed.

#### Validation Responses

In the case that a required field is missing from your request, the following response will be returned containing a unique ErrorCode and Message as well as the request body sent during the call. The possible ErrorCodes and Messages are listed below.

HTTP/1.1 400 Bad Request

Content-Type: application/json

{

"ErrorCode": "...",

"Message": "...",

"Data": {

“...Request Body...”

}

}

|  |  |  |
| --- | --- | --- |
| **ErrorCode** | **Message** | **Status Code** |
| AuthCaptureTransaction.OrderIDRequired | OrderID is required to authorize and capture a credit card. | 400 |
| AuthCaptureTransaction.AmountRequired | Amount is required to authorize and capture a credit card. | 400 |
| AuthCaptureTransaction.BuyerIDRequired | BuyerID is required to authorize and capture a credit card. | 400 |
| AuthCaptureTransaction.CardDetailsRequired | CardDetails.CreditCardID OR CardDetails.CardNumber and CardDetails.ExpirationDate are required to authorize and capture a credit card. | 400 |

#### Authorize.Net Error Responses

HTTP/1.1 200 OK

Content-Type: application/json

{

"transactionResponse": {

"responseCode": "3",

"authCode": "",

"avsResultCode": "...",

"cvvResultCode": "",

"cavvResultCode": "",

"transId": "0",

"refTransID": "",

"transHash": "...",

"testRequest": "0",

"accountNumber": "...",

"accountType": "",

"errors": [

{

"errorCode": "...",

"errorText": "..."

}

],

"transHashSha2": ""

},

"refId": "...",

"messages": {

"resultCode": "Error",

"message": [

{

"code": "...",

"text": "..."

}

]

}

}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **responseCode** | **errorCode** | **errorText** | **resultCode** | **message.code** | **message.text** |
| 3 | 5 | A valid amount is required. | Error | E00027 | The transaction was unsuccessful. |
| 3 | 6 | The credit card number is invalid. | Error | E00027 | The transaction was unsuccessful. |
| 3 | 7 | Credit card expiration date is invalid. | Error | E00027 | The transaction was unsuccessful. |
| 3 | 8 | The credit card has expired. | Error | E00027 | The transaction was unsuccessful. |

#### OrderCloud.io Error Responses

If an incorrect BuyerID was provided:

HTTP/1.1 404 Not Found

Content-Type: application/json

{

"Errors": [

{

"ErrorCode": "NotFound",

"Message": "Buyer not found: 1234",

"Data": null

}

]

}

If an incorrect CardDetails.CreditCardID was provided:

HTTP/1.1 404 Not Found

Content-Type: application/json

{

"Errors": [

{

"ErrorCode": "NotFound",

"Message": "Credit Card not found: 1234",

"Data": null

}

]

}

If an incorrect OrderID was provided:

HTTP/1.1 404 Not Found

Content-Type: application/json

{

"Errors": [

{

"ErrorCode": "NotFound",

"Message": "Order not found: 1234",

"Data": null

}

]

}