**Authorize.net integration documentation**

**JSON calls and responses**

**Call to the Authorize.net integration proxy**

**URL Action - POST**

**Test URL -** <https://api.ordercloud.io/v1/integrationproxy/authorizenettest>

**Headers**

Content-Type: application/json

Authorization: bearer {userToken}

**Body**

{

"BuyerID": "12345",

"OrderID": "99996",

"TransactionType": "authCaptureTransaction",

"Amount": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"CardholderName": "Bob Tom",

"PaymentID": null,

"CreditCardID": null,

"CardType": "visa",

"CardNumber": "4007000000027",

"ExpirationDate": "1220",

"CardCode": "001"

}

}

**Transaction Types**

1. **createChargeCreditCard**
   1. Use this function to create a new customer profile, a new credit card and charge that same card. This method will likely be used most often for first-time customers on your site. NOTE: This will simply authorize the charge, so you will need to complete priorAuthCaptureTransaction to capture those funds later.
2. **createCreditCard**
   1. Use this function to create a new credit card for an existing customer profile or create a new credit card and a new payment profile at the same time.
3. **updateCreditCard**
   1. Use this function to update a credit card for an existing customer profile. NOTE: if some fields in this request are not submitted or are submitted with a blank value, the values in the original profile are removed.
4. **deleteCreditCard**
   1. Use this function to delete a credit card from an existing customer profile.
5. **authCaptureTransaction** 
   1. Use this method to authorize and capture a credit card payment. A payment profile or a one-time credit card can be used.
6. **authOnlyTransaction** 
   1. Use this method to authorize a credit card payment. To actually charge the funds you will need to follow up with a capture transaction.
7. **priorAuthCaptureTransaction**
   1. Use this method to capture funds for a transaction that was previously authorized using authOnlyTransaction.
8. **refundTransaction**
   1. This transaction type is used to refund a customer for a transaction that was originally processed and successfully settled through the payment gateway. A refund can only be used for an amount less than the original order amount. To refund an entire order please use voidTransaction.
9. **voidTransaction**
   1. This transaction type can be used to cancel either an original transaction that is not yet settled or an entire order composed of more than one transaction. A Void prevents the transaction or the order from being sent for settlement. A Void can be submitted against any other transaction type.

\***Bold** denotes required fields.

**createChargeCreditCard**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**createChargeCreditCard**",

"**Amount**": 10,

"**CorporateCard**": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"**CardholderName**": "Bob Tom",

"PaymentID": null,

"CreditCardID": null,

"**CardType**": "visa",

"**CardNumber**": "XXXXXXXXXXXXXXXXXXX",

"**ExpirationDate**": "1220",

"**CardCode**": "001"

}

}

Response JSON

*Success*

{

"transactionResponse": {

"responseCode": "1",

"authCode": "EO96YM",

"transId": "20000682994",

"messages": [

{

"code": "1",

"description": "This transaction has been approved."

}

]

},

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error [Account already exists]*

{

"customerProfileId": "1200090911",

"customerPaymentProfileId": "1200044960",

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00039",

"text": "A duplicate customer payment profile already exists."

}

]

}

}

**createCreditCard**

Request JSON

{

"**BuyerID**": "12345",

"OrderID": "99996",

"**TransactionType**": "**createCreditCard**",

"Amount": 10,

"**CorporateCard**": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"**CardholderName**": "Bob Tom",

"PaymentID": null,

"CreditCardID": null,

"**CardType**": "visa",

"**CardNumber**": "XXXXXXXXXXXXXXXXXXX",

"**ExpirationDate**": "1220",

"CardCode": "001"

}

}

Response JSON

*Success*

{

"ID": "efgVo73eskqeZXVbVYM3mA",

"Editable": true,

"Token": "1200044901",

"DateCreated": "2016-05-19T15:08:41.803+00:00",

"CardType": "MasterCard",

"PartialAccountNumber": "0015",

"CardholderName": "Craig Biggio",

"ExpirationDate": "2020-12-01T06:00:00+00:00",

"xp": null

}

*Error [Customer Payment Profile Already Exists]*

{

"customerProfileId": "1200090911",

"customerPaymentProfileId": "1200044960",

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00039",

"text": "A duplicate customer payment profile already exists."

}

]

}

}

**updateCreditCard**

{

"**BuyerID**": "12345",

"OrderID": "99996",

"**TransactionType**": "**updateCreditCard**",

"Amount": 10,

"**CorporateCard**": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"CardholderName": "Bob Tom",

"PaymentID": null,

"**CreditCardID**": null,

"CardType": "visa",

"CardNumber": "XXXXXXXXXXXXXXXXXXX",

"ExpirationDate": "1220",

"CardCode": "001"

}

}

Response JSON

*Success*

{

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error [Missing required fields]*

{

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00003",

"text": "The 'AnetApi/xml/v1/schema/AnetApiSchema.xsd:cardNumber' element is invalid - The value '' is invalid according to its datatype 'String' - The actual length is less than the MinLength value."

}

]

}

}

**deleteCreditCard**

Request JSON

{

"**BuyerID**": "12345",

"OrderID": "99996",

"**TransactionType**": "**deleteCreditCard**",

"Amount": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"CardholderName": "Bob Tom",

"PaymentID": null,

"**CreditCardID**": null,

"CardType": "visa",

"CardNumber": "XXXXXXXXXXXXXXXXXXX",

"ExpirationDate": "1220",

"CardCode": "001"

}

}

Response JSON:

*Success*

{

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error [Credit card ID does not exist]*

*{*

*"ErrorCode": "Credit Card NotFound"*

*}*

**authCaptureTransaction**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**authCaptureTransaction** ",

"**Amount**": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"**CardholderName**": "Bob Tom",

"PaymentID": null,

"**CreditCardID**": null, **[If using a Saved card]**

"**CardType**": "visa", **[If using a One-time card]**

"**CardNumber**": "XXXXXXXXXXXXXXXXXXX", **[If using a One-time card]**

"**ExpirationDate**": "1220",**[If using a One-time card]**

"**CardCode**": "001" **[Both a saved card and a One-time card]**

}

}

Response JSON

*Success*

{

"ChargeStatus": "Approved",

"PaymentID": "1",

"Type": "CreditCard",

"DateCreated": "2016-05-20T14:37:16.667+00:00",

"CreditCardID": "vjU140a1S0ei4uLKXN8fvg",

"SpendingAccountID": null,

"Description": "Authorize.Net Payment",

"Amount": "5.0",

"xp": null

}

*Error [Order is fully paid]*

{

"Errors": [

{

"ErrorCode": "InvalidRequest",

"Message": "Cannot create Payment. Order is already fully paid.",

"StackTrace": null

}

]

}

**authOnlyTransaction**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**authOnlyTransaction** ",

"**Amount**": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"**CardholderName**": "Bob Tom",

"PaymentID": null,

"**CreditCardID**": null, **[If using a Saved card]**

"**CardType**": "visa", **[If using a One-time card]**

"**CardNumber**": "XXXXXXXXXXXXXXXXXXX", **[If using a One-time card]**

"**ExpirationDate**": "1220",**[If using a One-time card]**

"**CardCode**": "001" **[Both a saved card and a One-time card]**

}

}

{

"**buyerID**": "12345",

"**orderID**":"222222",

"**transactionType**":"**authOnlyTransaction**",

"**amount**": "60",

"**cardDetails**":{

"paymentID": null,

"**creditCardID**": "3WipnU4y\_EKAPnEuKnE0PA", **[If using a Saved card]**

"cardType": null,

"**cardNumber**": "1111111111111111", **[If using a One-time card]**

"**expirationDate**": "1220", **[If using a One-time card]**

"**cardCode**": "999" **[Both a saved card and a One-time card]**

}

}

Response JSON

*Success*

{

"ChargeStatus": "Approved",

"PaymentID": "1",

"Type": "CreditCard",

"DateCreated": "2016-05-20T14:40:32.623+00:00",

"CreditCardID": "vjU140a1S0ei4uLKXN8fvg",

"SpendingAccountID": null,

"Description": "Authorize.Net Payment",

"Amount": "5.0",

"xp": null

}

*Error (Order does not exist)*

{

"Errors": [

{

"ErrorCode": "NotFound",

"Message": "Order not found: 2",

"StackTrace": null

}

]

}

*Error (Duplicate Transaction)*

{

"transactionResponse": {

"responseCode": "3",

"authCode": null,

"avsResultCode": "P",

"cvvResultCode": null,

"cavvResultCode": null,

"transId": "0",

"refTransID": null,

"transHash": "027FF68F476BBAB653A5D24B807C01FA",

"testRequest": "0",

"accountNumber": "XXXX8888",

"accountType": "Visa",

"errors": [

{

"errorCode": "11",

"errorText": "A duplicate transaction has been submitted."

}

]

},

"refId": "1234",

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00027",

"text": "The transaction was unsuccessful."

}

]

}

}

**priorAuthCaptureTransaction**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**updateCreditCard**",

"**Amount**": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"CardholderName": null

"**PaymentID**": null, **[If using a Saved card or One-Time Card]**

"CreditCardID": null

"CardType": null

"CardNumber": null

"ExpirationDate": null

"CardCode": null

}

}

{

"**buyerID**": "12345",

"**orderID**":"222222",

"**transactionType**":"**priorAuthCaptureTransaction**",

"**amount**": "60",

"**cardDetails**":{

"**paymentID**":"1644874", **[If using a Saved card or One-Time Card]**

"creditCardID": null,

"cardType": null,

"cardNumber": null,

"expirationDate": null,

"cardCode": null

}

}

Response JSON

*Success*

{

"transactionResponse": {

"responseCode": "1",

"authCode": "PXKBT1",

"avsResultCode": "P",

"cvvResultCode": null,

"cavvResultCode": null,

"transId": "20000734192",

"refTransID": "20000734192",

"transHash": "FE917D2056AB5F92C69FE01C22CA4393",

"testRequest": "0",

"accountNumber": "XXXX1111",

"accountType": "Visa",

"messages": [

{

"code": "1",

"description": "This transaction has been approved."

}

]

},

"refId": "1",

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error (Payment already captured)*

{

"transactionResponse": {

"responseCode": "1",

"authCode": null,

"transId": "0",

"messages": [

{

"code": "311",

"description": "This transaction has already been captured."

}

]

},

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error (Incorrect Payment ID or Payment does not exist)*

{

"ErrorCode": "Payment NotFound"

}

**refundTransaction**

**NOTE: You cannot refund a transaction for the original amount. Refunds must be for less than the original amount, use Void to refund the full amount.**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**refundTransaction**",

"**Amount**": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"**CardholderName**": "Bob Tom",

"**PaymentID**": “754ufg7tutr87u”,

"**CreditCardID**": “jfhf7ryur7”, **[If using a Saved card]**

"CardType": null

"**CardNumber**": "XXXX", **[If using a One-time card, last four digits only]**

"**ExpirationDate**": "1220",**[If using a One-time card]**

"CardCode": null

}

}

Response JSON

*Success*

*Error (If attempting to refund full payment amount)*

{

"transactionResponse": {

"responseCode": "3",

"authCode": null,

"transId": "0",

"errors": [

{

"errorCode": "54",

"errorText": "The referenced transaction does not meet the criteria for issuing a credit."

}

]

},

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00027",

"text": "The transaction was unsuccessful."

}

]

}

}

*Error (Incorrect Credit Card Number)*

{

"transactionResponse": {

"responseCode": "3",

"authCode": null,

"transId": "0",

"errors": [

{

"errorCode": "6",

"errorText": "The credit card number is invalid."

}

]

},

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00027",

"text": "The transaction was unsuccessful."

}

]

}

}

*Error (Incorrect Credit Card Code)*

{

"transactionResponse": {

"responseCode": null,

"authCode": null,

"transId": null

},

"messages": {

"resultCode": "Error",

"message": [

{

"code": "E00003",

"text": "The 'AnetApi/xml/v1/schema/AnetApiSchema.xsd:cardCode' element is invalid - The value '66' is invalid according to its datatype 'AnetApi/xml/v1/schema/AnetApiSchema.xsd:cardCode' - The actual length is less than the MinLength value."

}

]

}

}

*Error (All required fields are not provided)*

*{*

*"Error": "buyerID, orderID, paymentID, and amount are required fields"*

*}*

*Error (Payment ID does not exist)*

*{*

*"Errors": [*

*{*

*"ErrorCode": "NotFound",*

*"Message": "Payment not found: RKaNZb65I0mbHj8E9UtxhA",*

*"Data": null*

*}*

*]*

*}*

**voidTransaction**

Request JSON

{

"**BuyerID**": "12345",

"**OrderID**": "99996",

"**TransactionType**": "**voidTransaction**",

"Amount": 10,

"CorporateCard": false,

"Assignments": {

"Users": [

{

"UserID": "725bh6"

}

],

"UserGroups": [

{

"UserGroupID": "725bh6"

}

]

},

"CardDetails": {

"CardholderName": null,

"**PaymentID**": “jsdfu78rfey7fh”,

"CreditCardID": null,

"CardType": "null,

"CardNumber": null,

"ExpirationDate": null,

"CardCode": null,

}

}

Response JSON

*Success*

{

"transactionResponse": {

"responseCode": "1",

"authCode": null,

"avsResultCode": "P",

"cvvResultCode": null,

"cavvResultCode": null,

"transId": "0",

"refTransID": "20000932346",

"transHash": "6C99F7D77D9F902D6C509D9D166F6D65",

"testRequest": "0",

"accountNumber": null,

"accountType": "Visa",

"messages": [

{

"code": "310",

"description": "This transaction has already been voided."

}

]

},

"refId": "1234",

"messages": {

"resultCode": "Ok",

"message": [

{

"code": "I00001",

"text": "Successful."

}

]

}

}

*Error (Transaction already voided)*

*{*

*"transactionResponse": {*

*"responseCode": "1",*

*"authCode": null,*

*"avsResultCode": "P",*

*"cvvResultCode": null,*

*"cavvResultCode": null,*

*"transId": "0",*

*"refTransID": "2261503493",*

*"transHash": "6C99F7D77D9F902D6C509D9D166F6D65",*

*"testRequest": "0",*

*"accountNumber": null,*

*"accountType": "Visa",*

*"messages": [*

*{*

*"code": "310",*

*"description": "This transaction has already been voided."*

*}*

*],*

*"errors": [*

*{*

*"errorCode": null,*

*"errorText": null*

*}*

*]*

*},*

*"refId": "1",*

*"messages": {*

*"resultCode": "Ok",*

*"message": [*

*{*

*"code": "{I00001}",*

*"text": "{Successful.}"*

*}*

*]*

*}*

*}*