## ## Overview

In some instances, you may want to allow a user to order on behalf of another user. We see this use case a lot in Customer Service Desk and Call Center scenarios where customers will call their orders in and the service rep places the order on the customer’s behalf. This workflow preserves the reporting data, email notifications, and presents the catalog ordering rules the buyer is configured for.

The OrderCloud.io API supports this capability by allowing certain users to make API calls on behalf of a buyer user, which we refer to as impersonation. If you're an admin user, with the `BuyerImpersonation` role you can impersonate any buyer user under your organizational umbrella as long as an applicable Impersonation Config has been created. If you're a buyer user with the `BuyerImpersonation` role can impersonate any other buyer user within the same buyer company as long as an applicable Impersonation Config has been created.

## ## Creating an Impersonation Config

ImpersonationBuyerID, ImpersonationGroupID, and ImpersonationUserID all reference the party you want to grant access to do the impersonating. While BuyerID, GroupID, UserID all reference the party who will be impersonated. The SecurityProfileID is the Security Profile you would like to grant the user doing the impersonating (only while they are impersonating). ClientID is the specific application impersonation will be allowed in, if you need to impersonate in multiple applications, you’ll need to set up multiple Impersonation Configs.

POST https://api.ordercloud.io/v1/impersonationconfig HTTP/1.1  
Authentication: Bearer put\_access\_token\_here

Content-Type: application/json

{

"ID": "",

"ImpersonationBuyerID": "",

"ImpersonationGroupID": "",

"ImpersonationUserID": "",

"BuyerID": "",

"GroupID": "",

"UserID": "",

"SecurityProfileID": "",

"ClientID": ""

}

## ## Retrieving the Access Token

After you have successfully created an applicable Impersonation Config, the next step is to retrieve that buyer user's Access Token by using the Users `GetAccessToken` Endpoint:

<div class="api-reference">API reference: [Get Access Token](https://documentation.ordercloud.io/api-reference#Users\_GetAccessToken)</div>

```

POST https://api.ordercloud.io/v1/buyers/{buyerID}/users/{userID}/accesstoken HTTP/1.1  
Authentication: Bearer put\_access\_token\_here

Content-Type: application/json  
  
{

"ClientID": "…",

"Roles": [

"DevCenterImpersonate"

]

}

```

## ## Subsequent Requests

The` access\_token` from the response will need to be included for each and every OrderCloud.io API request in the Authorization header prefaced by **`**Bearer**`**:

```

GET https://api.ordercloud.io/v1/buyers HTTP/1.1  
Authentication: Bearer put\_access\_token\_here

Content-Type: application/json

```

## ## Conclusion

You should now have a basic understanding of to impersonate a buyer user on Ordercloud.io, and when you may need to use this functionality.