

[Go to MightyCall](#)

Search for articles...

[All Collections](#)[Integrations](#)[How to integrate MightyCall with Salesforce through Zapier \(beta\)](#)

# How to integrate MightyCall with Salesforce through Zapier (beta)



Written by Karina

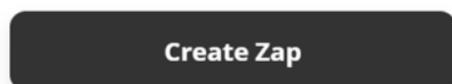
Updated over a week ago

- [Step 1. MightyCall's integration with Zapier. Sending information about a new call](#)
- [Step 2. Sending the call information to Salesforce](#)
- [Step 3. Creating a contact in Salesforce and adding it to MightyCall](#)

## Step 1. MightyCall's integration with Zapier. Sending information about a new call

For this step, you will be sending the information about the call from MightyCall to Zapier. During later steps, you will be sending this information straight to Salesforce.

1. Open [MightyCall's integration with Zapier](#).
2. Create a new Zap by clicking on the black button **Create Zap**.

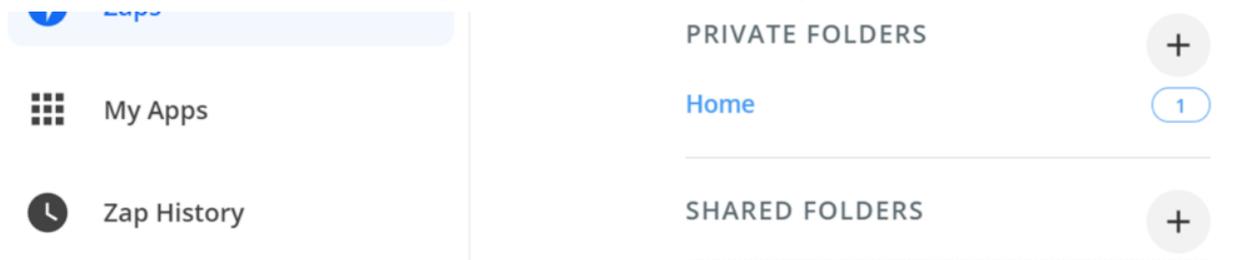


Dashboard



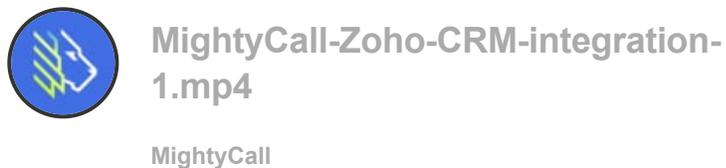
Zaps

## Zaps



3. In the search bar, enter MightyCall and add the application as the first action for your Zap.

4. Select a trigger **New Call**.



This trigger allows you to get information about a new incoming/outgoing call.

5. Click the Continue button, then click the **Sign In to MightyCall** button.

6. Log into your MightyCall account and copy the following information to Zapier:

- Your extension number (**My Profile** → **Extension**)
- API Key (**Settings** → **API integration** → **API Key for your account**)

7. Paste your extension and API key to Zapier and allow Zapier to access your MightyCall account. The link will be generated.



## MightyCall-Zoho-CRM-integration-2.mp4

MightyCall

00:45



8. Select the type of calls you want to send the CRM: incoming/outgoing. A separate Zap should be created for each call type. In the example below we have selected only *outgoing calls*.



## MightyCall-Zoho-CRM-integration-4.mp4

MightyCall

00:04



9. Use MightyCall to make an incoming or outgoing call, depending on what type of call you selected in the previous step. After that, you need to test the current step by clicking the **Test trigger** button.

A successful test result looks as follows. Press **Continue** if the result was successful.



## MightyCall-Zoho-CRM-integration-5.mp4

MightyCall



10. Click the [+] button.

- Choose app – MightyCall.
- Choose action event – Find Call Data.



## MightyCall-Zoho-CRM-integration-6.mp4

MightyCall



11. Choose account – select previously saved log in data.



## MightyCall-Zoho-CRM-integration-7.mp4

MightyCall



12. Set up action – select Call ID with a unique identifier of your call that provides information about the call.



## MightyCall-Zoho-CRM-integration-8.mp4

MightyCall



13. Test the current step by clicking on the **Test & Review** button. A successful test result looks as follows:

**Test action****Test was successful!**

We'll use this as a sample for setting up the rest of your Zap.

✓ A call was sent to MightyCall (1.0.0) about 1 minute ago.

Search call data...

**data:****id:** 4405e10a-a633-4f16-b557-0d790eef7ef5**caller:****name:****phone:** +1 (312) 762-2807**called:****1:****isConnected:** false**name:** N [redacted]**phone:** +1 (312) 762-2807**businessNumber:** +1 (312) 762-2807**dateTimeUtc:** 2021-07-19T10:58:01.821Z**duration:** 4**direction:** Incoming**callStatus:** Missed**callRecord:** null**isSuccess:** true

Now you can use the received call data for integration with various CRMs. Please don't close your Zap. You will need to add another step to this Zap for integration with your Salesforce CRM.

**Step 2. Sending the call information to Salesforce**

This integration allows us to save the call history in Salesforce with the binding of each call to a specific contact within the CRM.

1. Return to your created Zap.

- Click the [+] button to continue with the integration

• Add Event: Salesforce

- App Event – Salesforce
- Action Event – Find Record



## MightyCall-Salesforce-integration-1.mp4

MightyCall

00:11



2. Log into your Salesforce account through Zapier or the previously selected saved login details.

- Select Salesforce environment – Production. Click **Yes, Continue**.
- Confirm Zapier’s access to your Salesforce account by clicking on the **Accept** button.



## MightyCall-Salesforce-integration-2.mp4

MightyCall

00:17



3. Select the following actions:

- Salesforce Object – Contact

• Salesforce Object – Contact

- Field To Search By – Business Phone
- Search Value: for an incoming call – Caller Phone/for outgoing call – Called Phone



### MightyCall-Salesforce-integration-3.mp4

MightyCall

00:24 |



4. Additionally, you can configure the creation of a contact in the CRM if no records are found for the phone number specified in the previous step. To do this, check *Create Salesforce Record if it doesn't exist yet?*

Fill in the fields as follows:

- Last Name – From MightyCall
- Business Phone: for incoming call – Caller Phone/ for outgoing call – Called Phone

The other fields are not mandatory, so you can skip them.



### MightyCall-Salesforce-integration-4.mp4

MightyCall

00:33 |



5. Press **Continue** and then test the step by clicking the **Test & Review** button.



### MightyCall-Salesforce-integration-5.mp4

MightyCall

00:10 |



Don't close your Zap.

Now you need to add the call information to the contact.

6. Click the [+] button to continue with the integration.

- Choose app – Salesforce
- Action event – Create Record



### MightyCall-Salesforce-integration-6.mp4

MightyCall

00:11 |

7. At the login step, select the previously saved data and press **Continue**.

8. Fill in the fields as follows:

- Salesforce Object – Task
- Name → Custom → *Find or Create Record in Salesforce* – ID
- Subject – the subject of our call. You can combine information from the previous steps.
- Due Date Only – today
- Status – Completed
- Call Duration – Call Duration
- Call type – Incoming or Outbound (based on your previous selection for this Zap)
- Task Subtype – Call



**MightyCall-Salesforce-integration-7\_1.mp4**

MightyCall

00:45 |

The other fields are not mandatory, so you can skip them.

9. Test this step. The result will be the creation of a call record in your Salesforce account, tied to the contact found/created.

Turn on your Zap.



# MightyCall-Salesforce-integration-8.mp4

MightyCall

00:08

A new call in Salesforce will look as follows in the contacts:

The screenshot shows a Salesforce contact record for a contact named "From MightyCall". The contact details include a name, account name, title, type, address information, and additional information like fax, home phone, department, lead source, and advertisement. The right-hand side of the record shows an activity log with a single entry: "Outgoing call with +1 [redacted] You logged a call". The activity log also includes filters for "All time", "All activities", and "All types", and options to "Refresh", "Expand All", and "View All".

When clicking on the call:

The screenshot shows a Salesforce task record for a task named "Outgoing call with +1 [redacted]". The task is marked as "Completed" and has a status of "Completed". The task details include a name, related to, and a status. The task is also associated with a contact record for "From MightyCall".

Task Information	
Assigned To	Related To
 <a href="#">MightyCall</a>	
Subject	Name
Outgoing call with +1	<a href="#">From MightyCall</a>
Comments	
Due Date	
8/10/2021	

### Step 3. Creating a contact in Salesforce and adding it to MightyCall

This integration allows us to automatically create a contact in MightyCall after we create it in Salesforce.

#### 1. Create a new Zap.

- App – Salesforce
- Trigger event – New record



MightyCall-Salesforce-integration-9.mp4

MightyCall

00:08



#### 2. Log in to Salesforce using your credentials.

3. Select Salesforce environment – Production. Click **Yes, Continue**. Allow Zapier to access your CRM data by clicking **Allow**.



MightyCall-Salesforce-integration-10.mp4

MightyCall

00:18



4. Set up trigger.

- Salesforce Object = Contacts



MightyCall-Salesforce-integration-11.mp4

MightyCall

00:08



5. Create a test contact in Salesforce: First name, Last name, phone number, and email.

6. Go back to Zapier and test the current step by clicking the **Test Trigger** button. The result should look as follows.



MightyCall-Salesforce-integration-12.mp4

## MightyCall

00:06



7. Click the [+] button to continue with the integration.

8. Choose app – MightyCall.

- Action event – Find or Create Contact and click **Continue**.



**MightyCall-Salesforce-integration-13.mp4**

MightyCall

00:08



9. Now log into your MightyCall account.

- Copy your extension (Profile – Personal details – Extension)
- Copy your API key (**Settings** → **API integration** → **API Key for your account**).

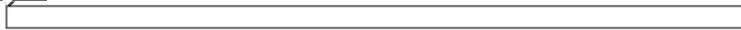
Or select previously saved credentials.



## MightyCall-Salesforce-integration-14.mp4

MightyCall

00:05



10. Add a Set up action.

- Phone number – Phone



## MightyCall-Salesforce-integration-15.mp4

MightyCall

00:08



11. Check the box *Create MightyCall Contact if it doesn't exist yet?*

Fill in the following fields:

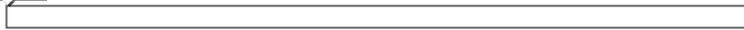
- First Name – First Name
- Last Name – Last Name
- Email – Email



## MightyCall-Salesforce-integration-16.mp4

MightyCall

00:26



12. Press **Continue** and then test the following step by clicking **Test & Review**.  
A successful result looks as follows:



## MightyCall-Salesforce-integration-17.mp4

MightyCall

00:08



13. Go to MightyCall and check the new contact.

14. Turn on the created Zap.

After enabling the Zap, whenever you create a new contact in Salesforce, the contact's data will be automatically transferred to MightyCall. For each subsequent incoming/outgoing call, that contact will be identified.

Did this answer your question?

