

# OpenAI Integration Setup

What's in this article:

- [Background Information](#)
- [How to set up your MSPbots integration with OpenAI](#)
- [How to verify the connection between MSPbots and ChatGPT](#)
- [OpenAI API docs](#)
- [Why MSPbots is Unable to Return Expected Message After Successful Connection to OpenAI Integration](#)

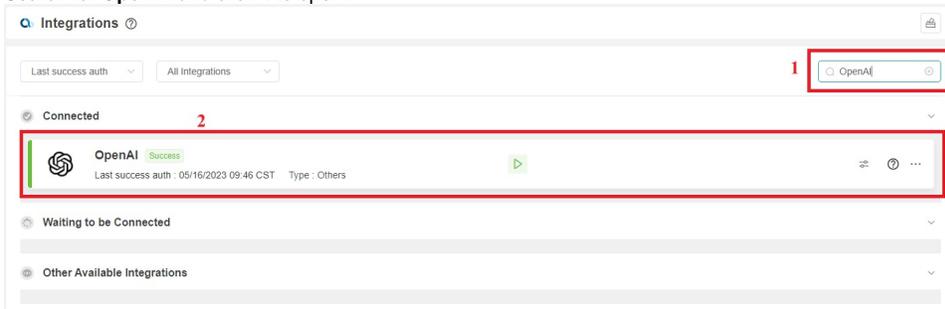
## Background Information

- The messages you send on MSPbots have two types:
  - Business logic commands - These messages include clocking in and out, break, etc.
  - Non-business logic commands - These are unknown commands, such as "How are you".
- The responses you receive on MSPbots are classified into the following:
  - When you successfully connect, MSPbots will directly give the following types of responses:
    - For business logic commands, MSPbots will directly respond to those commands.
    - For unknown commands, such as "How are you". For these commands, you will receive a response from ChatGPT on MSPbots because MSPbots connects with ChatGPT by setting the API interface, server address, port, and other parameters to call ChatGPT and establish a connection. ChatGPT will then send its reply to MSPbots.
  - When MSPbots responds with "You can also connect to ChatGPT," it means that you have not connected to OpenAI and need to connect to ChatGPT. Refer to [How to set up your MSPbots integration with OpenAI](#) for instructions on how to integrate your software.
  - When MSPbots returns "ChatGPT: xxxxxx," where "xxxxxx" is the error message returned by ChatGPT, it may mean that the connection has failed or there is an error. For corresponding explanations, read <https://platform.openai.com/docs/guides/error-codes>.
- The section below provides a guide on how to successfully connect MSPbots to OpenAI.

## How to set up your MSPbots integration with OpenAI

To connect MSPbots to OpenAI:

1. Create the OpenAI API Key.
  - a. Register an account with OpenAI.
  - b. Log in and enter your [account on the API keys page](#).
  - c. Click the **Create new secret key** button and click **Copy** to save the key locally.
  - d. [Add a billing limit](#) to maintain usage within a manageable range and to avoid unforeseen or excessive charges.
2. Test your connection. If you encounter an error, check if the OpenAI account you used to log in has the correct credentials.
3. Next, go to the MSPbots app and navigate to [Integrations](#).
4. Search for **OpenAI** and click it to open.



5. Enter the **API Key**, then click **Save And Sync**.

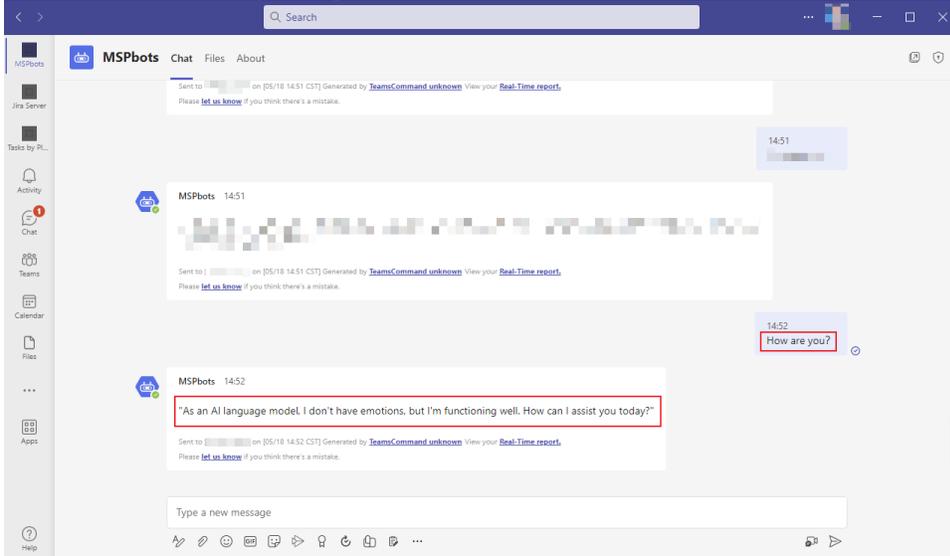


## How to verify the connection between MSPbots and ChatGPT

Follow these steps to test if the connection between MSPbots and ChatGPT is successful:

1. Enter a non-business command in MSPbots, such as "How are you?".

2. If MSPbots responds with any message, then the successful connection between MSPbots and ChatGPT is confirmed.



## OpenAI API docs

- You can view the documentation for OpenAI API at <https://platform.openai.com/docs/api-reference/introduction>
- If you want to learn about OPEN AI's terms of use and privacy policy for users, please refer to:
  - [Terms & policies](#)
  - [Privacy policy](#)

## Why MSPbots is Unable to Return Expected Message After Successful Connection to OpenAI Integration

We apologize for the inconvenience. The reason why MSPbots is unable to provide the expected response is because the unknown command you sent belongs to the following filtered topics.

- **Project management**
- **Weather status/forecast**
- **Writing advertisement campaign/ video script**
- **Tell me a joke**
- **Write an email introducing myself**
- **Recipes**

MSPbots uses OpenAI to respond to unknown commands. However, due to the mandatory limitations of the Microsoft Teams App, they require us to add filters to the messages returned by OpenAI for topic filtering. They believe that MSPbots should stay focused on supporting designated use cases. If user inputs are not related to the intended purpose of we (For e.g. Jokes, recipes), we should guide the user back to supported topics and prevent the user from engaging in unrelated discussions. Therefore, Teams will only approve the integration of the OpenAI application into Teams if these topics pass the filtering process.

Hi Team,

Thanks for your response. Regarding the issue#11 (Topic filtering) please find the below guidelines and Teamsubm recommendations to fix the issue

- Ensure app does not engage unrelated topics instead provides a graceful failure message and way forward to use designated functionality.

For example: If user requests below topics, the bot must not provide the correct response instead it should provide a grace full message for the user to request/ask about the topics related to the app functionality.

1. Project management
2. Weather status/forecast
3. Writing advertisement campaign/ video script
4. Tell me a joke
5. Write an email introducing myself

Thank you,  
VM  
Microsoft Teams App submissions (Teamsubm)

Hi ,

Thank you for comments. As per the latest Microsoft guidelines the given filter: **For e.g. jokes, recipes** in the AI bot need to be filtered and it is a mandatory fix to approve the application. For issue#13 we have re-validated the app after re-registering. And here we confirm to you that it is working fine and good to close it. from our end.

Thank you,  
VS  
Microsoft Teams App submissions (Teamsubm)

In that case, when you send an unknown command related to the mentioned topics, you will only receive the message: **As an AI language model, I can help you with various tasks such as answering questions, providing information, generating text, summarizing articles, and more. Let me know what you need help with specifically and I'll do my best to assist you.**

4:07 PM

tell me a joke

MSPbots 4:07 PM



As an AI language model, I can help you with various tasks such as answering questions, providing information, generating text, summarizing articles, and more. Let me know what you need help with specifically and I'll do my best to assist you.

AI-generated content may be incorrect.

Sent to [redacted] on [11/17 16:07 CST] Generated by [TeamsCommand unknown](#) View your [Real-Time report](#).

Please [let us know](#) if you think there's a mistake.