How to Customize the Commands Used in Chat

You can use modified commands in MS Teams to return specific information from the MSPbots apps. This is done by configuring the Message Command bot block to recognize custom commands that will trigger your bot and return needed information.

What's on this page

- [What is the Message Command bot block? Why should I modify it?](#)
- [Relevant applications and permissions](#)
- [How to configure the Message Command bot block](#)
- [MS Teams Command Template Bots](#)
    - [Related topics](#)

What is the Message Command bot block? Why should I modify it?

The Message Command bot block is where chat commands for the Attendance Manager, NextTicket Manager, and Utilization apps are configured to return specific information. This block recognizes preset commands that, when sent to MS Teams or Slack, return needed information according to the set schedule or on demand.

The preset commands for the MSPbots template bots are customizable in case none apply to your business requirements. If you have specific business needs that are not currently addressed by our preset commands, you can customize commands in the Message Command bot block.

For example, if you prefer to use the **check in** command instead of **in** for recording attendance in MS Teams, you can create a new bot, set up the **check in** command in the Message Command block, and then add corresponding Attendance-in and Alert blocks so that entering the command **check in** in MS Teams will automatically trigger the attendance bot you created.

Only MS Teams allow the addition of new commands. Right now, Slack recognizes the commands nt and help only, but will soon recognize new commands in the upcoming feature releases.

Relevant applications and permissions

This information applies to MS Teams, Slack, Attendance Manager, NextTicket Manager, and Utilization apps. Only users with admin permissions can create bot blocks and configure the Message Command block. If you do not have the required permission, contact your administrator for help.
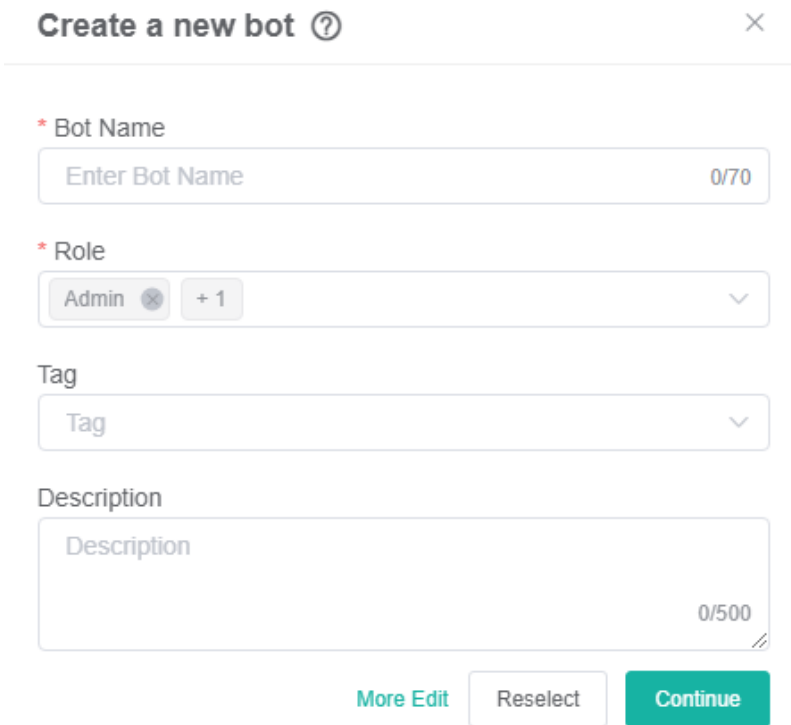
How to configure the Message Command bot block

The steps below show how you can create custom commands in the Message Command bot block. For this example, we want to set up a bot that will use the command **check in** instead of the preset **in** command. Do the following:
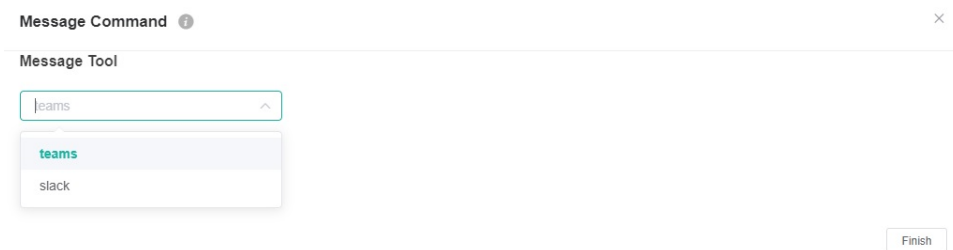
1. Log in to the **MSPbots app** using your Office 365 account.
2. Navigate to **Bots** on the left menu then click the **New Bot** button on the Bots screen.
3. When the Create a new bot window opens, select **Start From Blank** > **Message Command**.
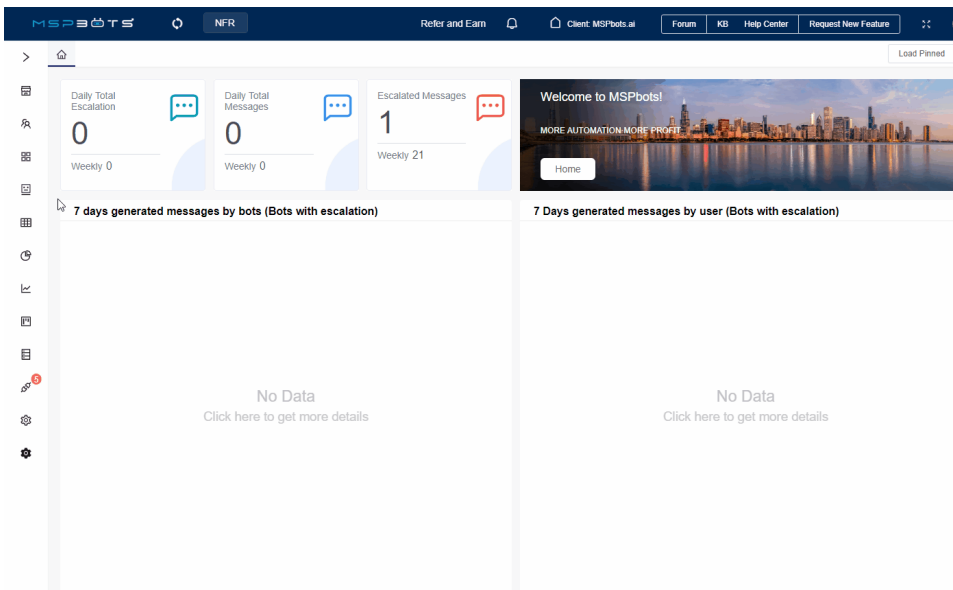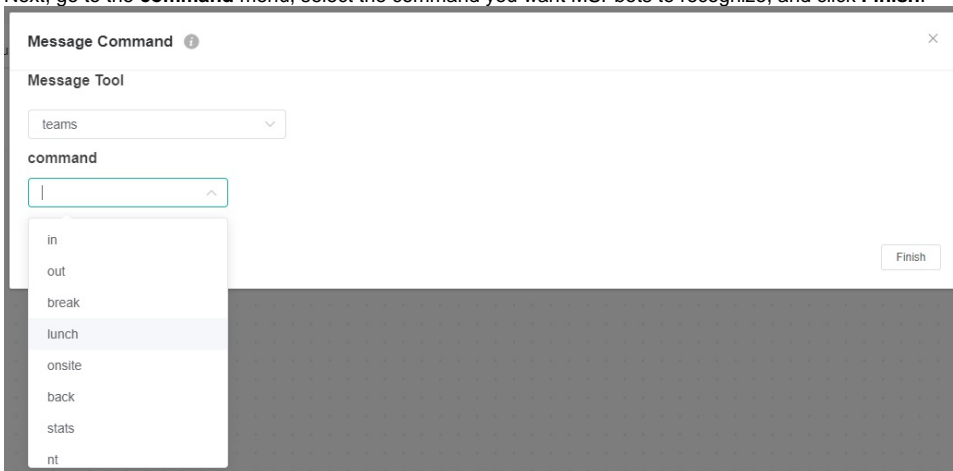
4. When the next Create a new bot window appears, fill in the required fields.

Create a new bot ⑦   ✕

* Bot Name

| Enter Bot Name   0/70 |

* Role

| Admin ⊗   + 1   ⌄ |

Tag

| Tag   ⌄ |

Description

| Description |
| |
| 0/500 |

More Edit   [Reselect]   [**Continue**]

    a. Enter a name in the **Bot Name** field.
    b. Identify the **Role** that will have access to the bot.
    c. Add a clear **Description** of the bot. (Recommended)
    d. Then click **Continue** to open the Message Command pop-up window.

5. In the Message Command pop-up window, click the **Message Tool** dropdown list and select a tool. To use MS Teams to trigger your bot, select **Teams**; to use **Slack**, select Slack instead.
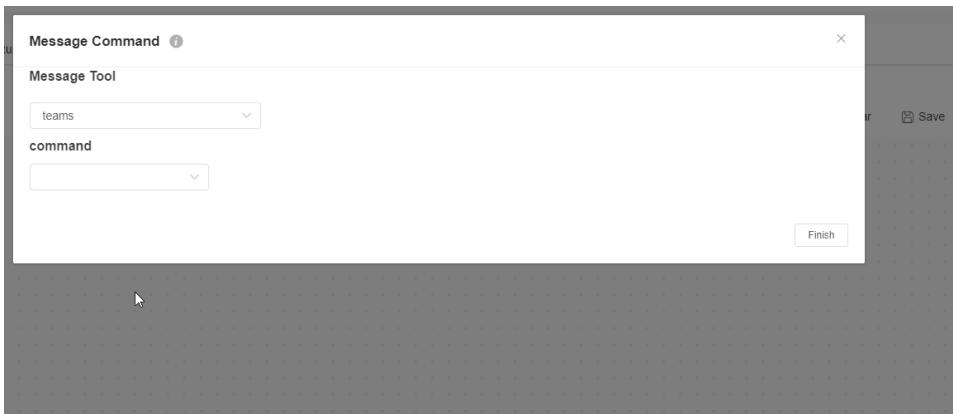
Message Command ⓘ   ✕

Message Tool

| teams   ⌃ |

| **teams** |
| slack |

[Finish]

6. Next, go to the **command** menu, select the command you want MSPbots to recognize, and click **Finish**.

Message Command ⓘ   ✕

Message Tool

| teams   ⌄ |

command

| | ⌃ |

| in |
| out |
| break |
| lunch |
| onsite |
| back |
| stats |
| nt |

[Finish]

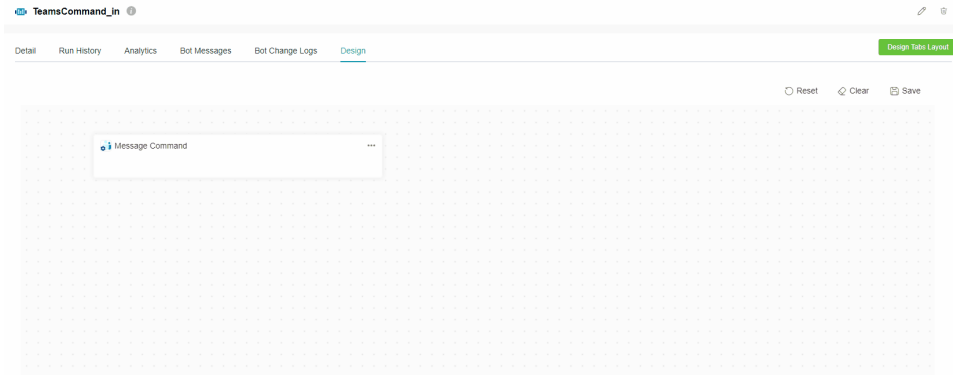For more information on the available commands for MS Teams, read the section on Teams Command Template Bots.

To use your preferred command, enter your custom command then click **Finish**. For example, if you want to trigger the attendance bot using the **check in** command instead of the **in** command, enter **check in**.

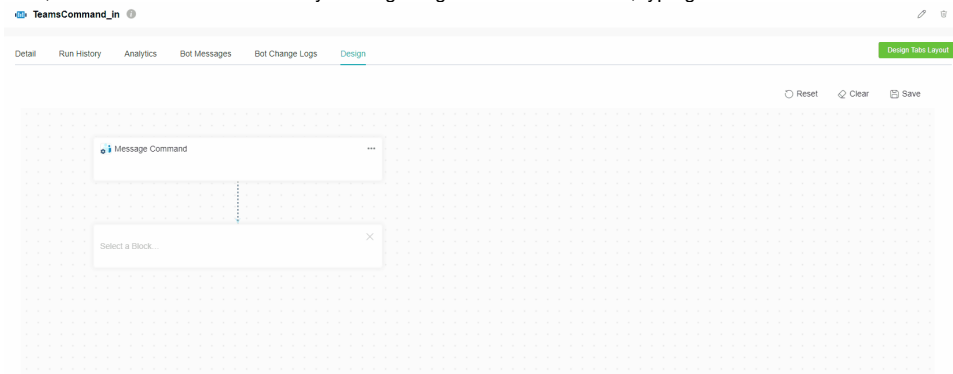*Type commands in lowercase when adding new commands. Example: Type check in instead of CHECK IN.*

*The feature to add custom commands is only supported in MS Teams for now and is not yet available in Slack. Only the **nt** command and **help** commands are currently recognized by Slack.*
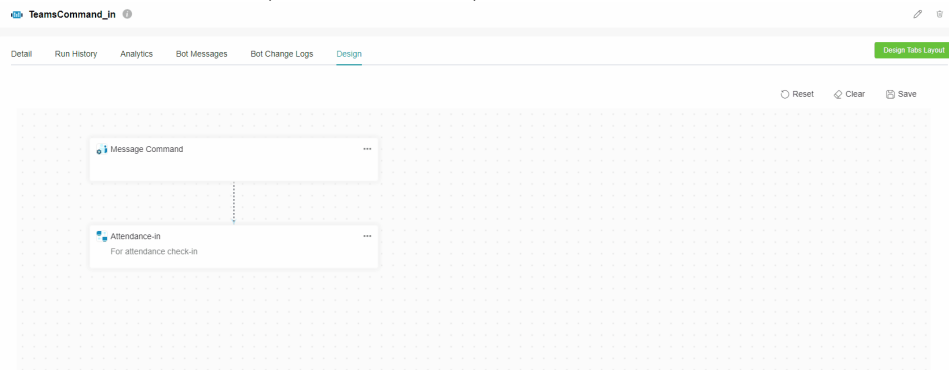
7. Click on the **Message Command** block, then click ⊕ under it, and select **Add a Block**.

8. Next, add an Attendance-in block by clicking the generated blank block, typing **Attendance-in** in the search bar, and selecting **Attendance-in.**
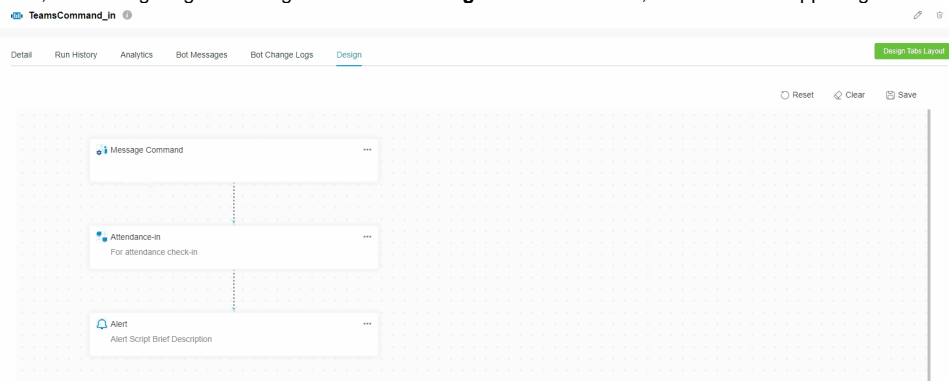
9. Click the **Attendance-in** block, then click ⊕ under it, and select **Add a Block**.

10. Next, add an Alert block by clicking the generated blank block, typing **Alert** in the search bar, and selecting **Alert**.
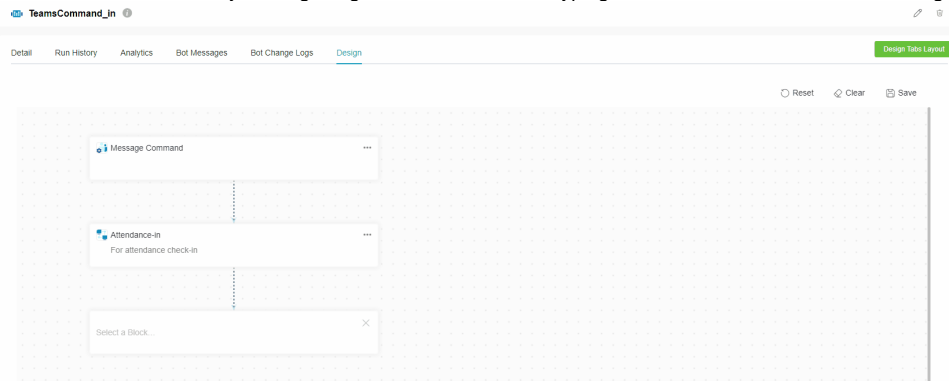
11. Next, start configuring the settings. Go to the **Message Command** block, click ••• in the upper-right corner of the block, and select **Edit**.

12. Verify if the selections for Message Tool and command are the same as your selections in Steps 4 and 5. Click **Next** once confirmed.

13. When the Attendance-in window opens, customize the responses for each of the following scenarios:
    - Clock in - Success
    - Clock in time is outside working hours -or- Clock in time is incorrect
    - Clock in update - Success
    - Wrong command

- Clock in is done by an invalid account

14. When done, click **Next** to proceed to the Alert configuration window.
15. Follow the prompts on the Alert window and configure the Alert block according to your preferences.

16. Click **Finish**.

17. Click **Save** on the upper right corner of your bot's window.
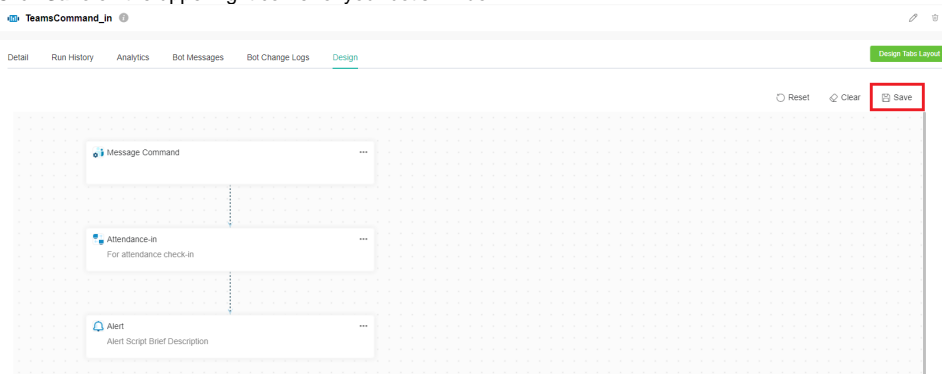
18. Verify your settings. Go to MS Teams and type **check in**. The command should be recognized and return a response based on the configurations you made.

MS Teams Command Template Bots

The table below displays the commands available for MS Teams, including their descriptions, configuration blocks, and corresponding need for alerts. Use this table as a guide when modifying message commands.

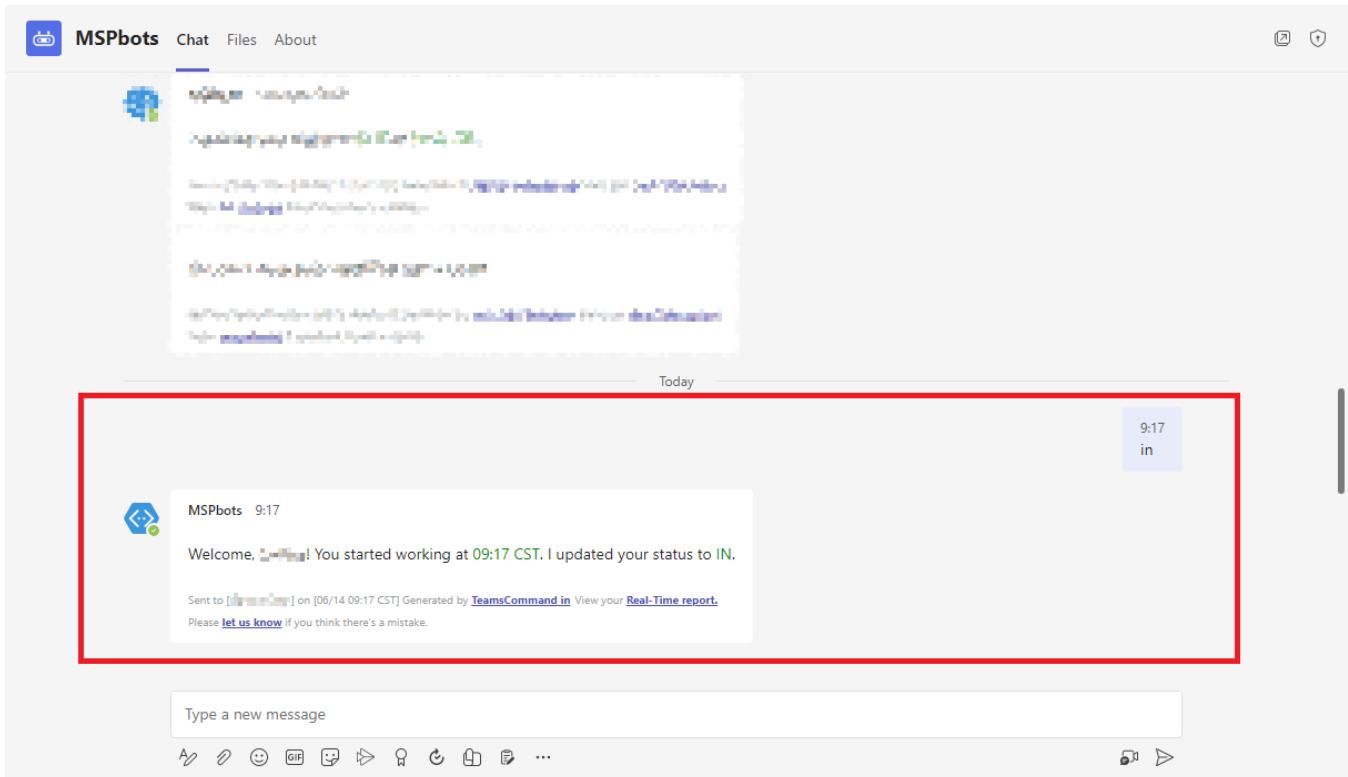| Template Bots	Message Command	Corresponding Block	Need for Alert Block	Descriptions
TeamsCommand help	**hi**	Command-Help	Yes	Bot that recognizes the **hi** command to return answers related to help information
TeamsCommand help	**hello**	Command-Help	Yes	Bot that recognizes the **hello** command to return answers related to help information
TeamsCommand help	**help**	Command-help	Yes	Bot that recognizes the **help** command to return answers related to help information
TeamsCommand in	**in**	Attendance-in	Yes	Bot that recognizes the **in** command for checking in into the Attendance Manager
TeamsCommand out	**out**	Attendance-out	Yes	Bot that recognizes the **out** command when logging off in the Attendance Manager
TeamsCommand	**break**	Attendance-break	Yes	Bot that recognizes the **break** command for logging a pause from work in the Attendance

nd break				Manager
TeamsCommand lunch	**lunch**	Attendance-lunch	Yes	Bot that recognizes the **lunch** command for logging a lunchbreak the Attendance Manager
TeamsCommand onsite	**onsite**	Attendance-onsite	Yes	Bot that recognizes the **onsite** command for the Attendance Manager check-in
TeamsCommand back	**back**	Attendance-back	Yes	Bot that recognizes the **back** command when logging back to work in the Attendance Manager
TeamsCommand pto	**pto**	Attendance-pto	Yes	Bot that recognizes the **pto** command to handle related leave operations
TeamsCommand_wfh	**wfh**	Attendance-wfh	Yes	Bot that recognizes the **wfh** command for the Attendance Manager check-in
Tech Stats Reviewer	**stats** or **qa**	Utilization-stats	Yes	This bot reports the total working hours, utilization rate, overlapping time entries, ongoing work orders, and pending tasks before the end of each user's workday.
TeamsCommand nt	**nt**	NT command message formatting	Yes	When used, the **nt** command the bot returns data in a custom format.
TeamsCommand reject	**reject**	Teams Command-reject	No	Bot that recognizes the **reject** command to skip the current ticket. This bot needs to be used with the **nt** command.
TeamsCommand unknown	**unknown**	Teams Command-ChatGPT	Yes	Bot that returns answers from ChatGPT. You need to set up your OpenAI integration before you can start using this bot.

Related topics

- How to Configure the REST API Call Bot Block
- How to Create or Modify Halo Tickets using a Rest API-based Bot