

How to Create or Modify Halo Tickets using a Rest API-based Bot

What's in this page:

- [Prerequisites](#)
- [Required Permissions](#)
- [How to Create a Bot to Create or Modify Halo Tickets](#)
 - [How to view the ID for ticket status or ticket type](#)
- [Related Topics](#)

Prerequisites

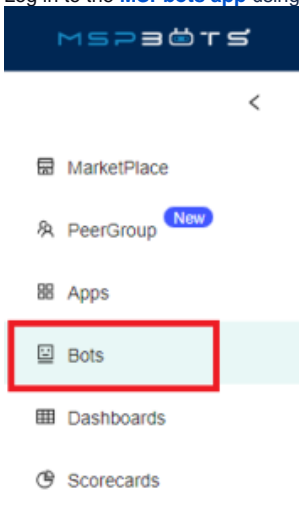
- (Optional) When you need to **update** tickets, it is crucial to clarify the following before proceeding with the configuration:
 - Determine which tickets meet the criteria for modification.
 - Due to Halo supports modifications for over 200 attribute fields, please specify which parameters and attribute names you want to modify for these tickets. Please refer to <https://halo.haloservicedesk.com/apidoc/resources/tickets>.
- (Optional) When you need to **create** tickets, it is essential to clarify the following before proceeding with the configuration:
 - Identify the field parameters and attribute names for the tickets you are creating. Please refer to <https://halo.haloservicedesk.com/apidoc/resources/tickets>.
- Create a widget or dataset to filter the data needed for the Halo tickets you want to create or modify.

Required Permissions

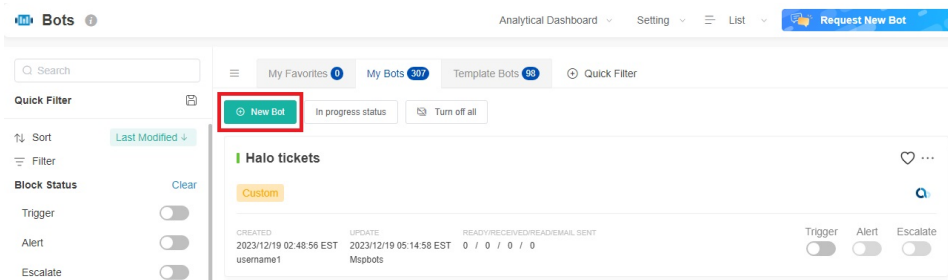
If you need to configure this guide, you need to have **admin** permissions.

How to Create a Bot to Create or Modify Halo Tickets

1. Log in to the [MSPbots app](#) using your Office 365 account and go to [Bots](#).



2. Click **New Bot**.



3. In the Start From Blank tab, select **Trigger**.


Create a new bot ⓘ

Start From Blank


Starting from blank lets you create a totally custom flow to meet your needs.

Start From Bot Template


Fully functional bots make this the easiest way to start, plus you still can add customizations.




Trigger



Real-time Data



Message Command



New Integration Trigger

4. In the **Create a new bot** popup, fill in the following fields:

- **Bot Name**
- **Role**
- **Tag**
- **Description**

Create a new bot ⓘ

* Bot Name

0/70

* Role

Admin ⓘ

+ 1

▼

Tag

▼

Description

0/500

Reselect

Continue

5. Click **Continue**.

6. On the bot page, click **Design**.

Halo tickets

Detail

Run History

Analytics

Bot Messages

Bot Change Logs

Design

Halo tickets

Trigger

Alert

Escalate

Recent run history

All runs

Start	Duration (s)	Status
No Data		

7. Configure the Trigger block, please refer to [2. Set up the bot trigger](#).

- For **I want the bot to trigger when**, add the widget you have created as the data source.
- For **meets the following criteria**, if you want to further filter the data, you can configure the filtering criteria here.
- For **I want the bot to run based on this schedule**, set to repeat every 3 hours.

Due to the rate limit imposed by the Halo app, the ticket synchronization frequency is as follows:

- ✓ If you have configured the WebHook ,
 - During working hours, all tickets are synchronized through WebHook , and MSPbots do not actively synchronize them.
 - During non-working hours, the synchronization occurs once every 2 hours.
- If you have not configured the WebHook ,
 - During working hours, the synchronization occurs every 5 minutes.
 - During non-working hours, the synchronization occurs once every 2 hours.

Therefore, in order to avoid duplicate API calls, please set the triggering frequency to every 3 hours.

Trigger

I want the bot to trigger when

Widget Halo / Halo Rest API Ticket

meets the following criteria:

AND OR

Select Fields

I want the bot to run based on this schedule:

America/Chicago

Starting at

2023-12-20

11:28:38

Repeat every

3

Hour

Repeat every

1

minutes

from hour

to


on day of week

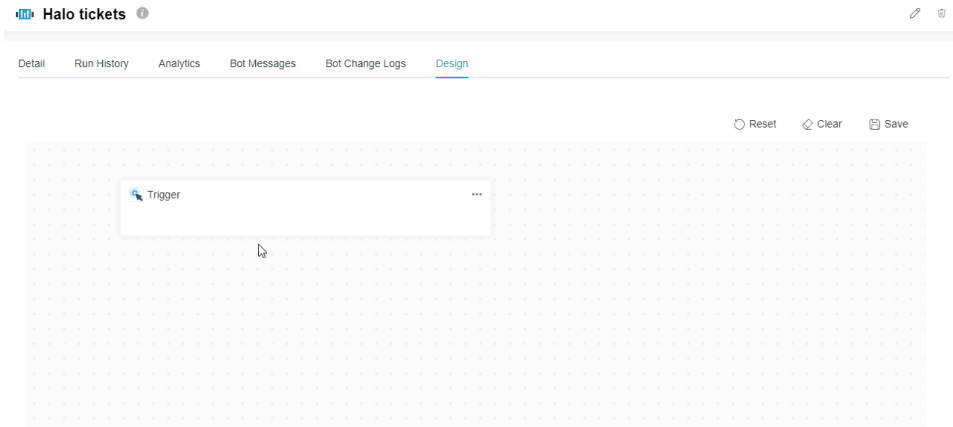
Select

Advanced Scheduler

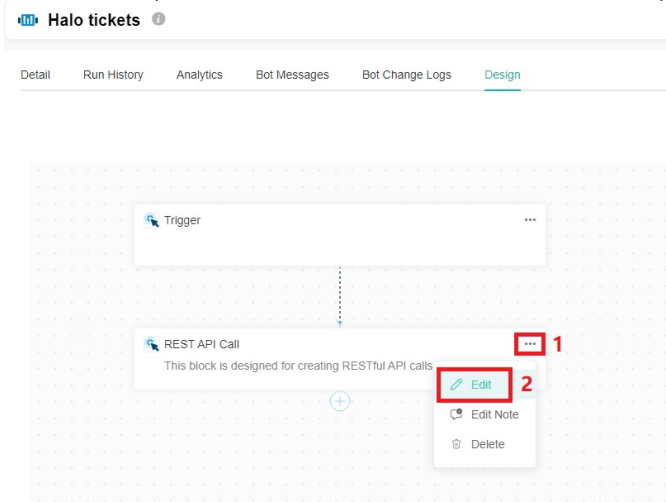
Setting

Next

8. Add **REST API Call** block - Click Trigger block, click  the plus button, click Add a block, click **Add a block**, search for **REST API Call** in the search bar, and click that block to add it.



9. Click  the ellipsis button, select **Edit** to enter the REST API Call block page.




10. On the **REST API Call** block page, fill in the following fields:
- url - Fill in the interface address for creating or modifying tickets in Halo: **https://{host}/api/Tickets**, for example: <https://mspbotstest.halopsa.com/api/Tickets>.
 - integration - Select **Halo** from the dropdown menu. The program has already been authorized for Halo automatically, so no additional action is required.
 - method - Select **POST** from the dropdown menu. For specific details, please refer to the API documentation. The POST method is used for creating and updating tickets.
 - headers - Can be left empty.
 - params
 - Click **body**.
 - Select **json**.
 - Input parameter fields in the JSON body, using the following columns as an example.

```
[
  {
    "actioncode":0,
    "dateoccurred":"2023-12-15T14:35:55.618Z",
    "summary":"{job_title}",
    "details":"{user_name}",
    "status_id":"2",
    "tickettype_id":"{update_source}"
  }
]
```

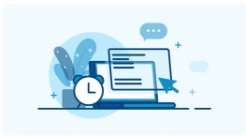


1. About the example:

- The "actioncode", "dateoccurred", "summary", "details", "status_id", "tickettype_id" are from <https://halo.haloservicedesk.com/apidoc/resources/tickets>.
- The {job_title}, {user_name}, {update_source} are from the data you added in the widget or dataset in the Trigger block. Please note that the format of the field must be **{Field}**.
 - To view the field parameters:

1. Locate the **meets the following criteria** in the **Trigger** block and click .

Trigger ⓘ




Define the bot and the conditions for when it will execute.

I want the bot to trigger when

Dataset Teams / Office 365 AD users

meets the following criteria:

AND OR 

user_name Equals @mspb

I want the bot to run based on this schedule: Asia/Shanghai

☒ Starting at 2022-11-09 03:33:03

Repeat every 3 Hour

☐ Repeat every 10 minutes

from hour 00:00 to 23:00 on day of week Monday

☐ Advanced Scheduler 0 0 3/5 * * ? *

2. View the fields in the Row Data.

Data Inspector

Raw Data

SQL

Field Description

Query

Error Log

gender	user_name	real_name	update_source	work_schec
	@mspbots.ai		1	true

« < 1 > » 10 Items per page 1 0

- 2. Every data filtered by widgets will trigger one bot execution.
 - 3. The field values enclosed in {} in the JSON body will be dynamically replaced with the values of the queried data.
 - 4. Please refer to the input parameter fields here: <https://halo.haloservicedesk.com/apidoc/resources/tickets>.
- Here are some commonly used attribute field explanations.

Field	Description
id	The ID of the ticket.

dateoccurred	The creation time of the ticket.
summary	A brief overview of the ticket.
details	Further information regarding the ticket.
status_id	The ID number corresponding to the status of the ticket. For example, 1 for New, 2 for In Progress. If you want to know the ID for each status, please refer to How to view the ID for ticket status or ticket type .
tickettype_id	The ID number corresponding to the ticket type. For example, 1 for Incident, 21 for Alert. If you want to know the ID for each ticket type, please refer to How to view the ID for ticket status or ticket type .

11. Click **Finish** to complete the configuration of the block.

http client

REST API Call

uri: https://mspbtest.halopsa.com/api/Tickets

integration: Halo

method: post

headers:

key	value
No Data	

query

body

none

form-data

x-www-form-urlencoded

json

xml

params:

```
{
  "actioncode":0,
  "dateoccurred":"2023-12-15T14:35:55.618Z",
  "summary":"{job_title}",
  "details":"{user_name}",
  "status_id":"2",
  "tickettype_id":"{update_source}"
}
```

Previous

Finish

12. After configuring, click the **Save** button to save the Bot.

Halo tickets

Detail

Run History

Analytics

Bot Messages

Bot Change Logs

Design

Design Tabs Layout

Reset

Clear



Save

Trigger

REST API Call

This block is designed for creating RESTful API calls.

13. Click **Detail**, open the Bot's runtime switch and wait for the scheduled task to trigger. After the Bot is running, it will create new tickets or modify existing tickets based on your configuration.

 **Halo tickets** 

Detail

Run History


Analytics

Bot Messages

Bot Change Logs

Design

Halo tickets



☒ Trigger

☒ Alert

☐ Escalate

Recent run history

[All runs](#)

Start	Duration (s)	Status
No Data		

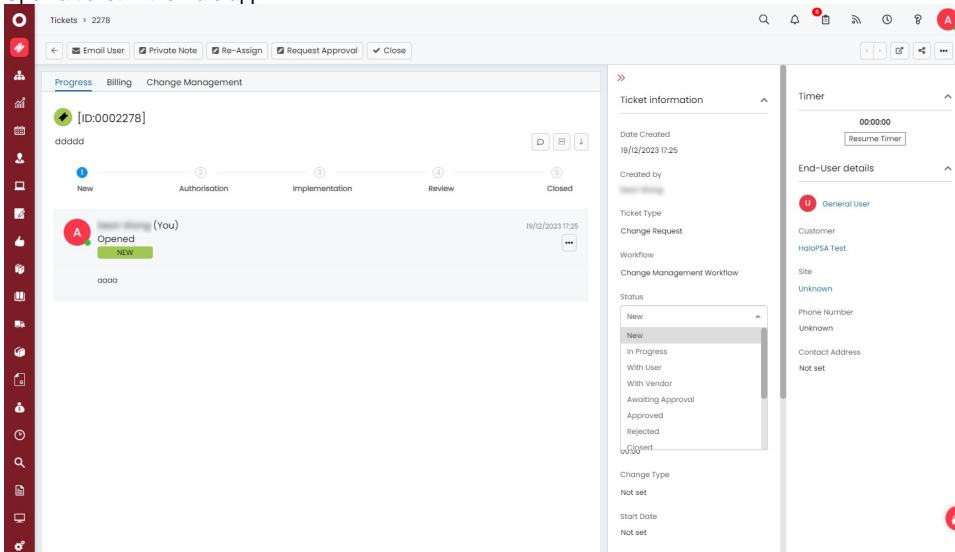
How to view the ID for ticket status or ticket type

**Viewing the ID for ticket status and ticket type follows a similar process. The following guide provides an example of how to view the ID for the "In Progress" ticket status.*

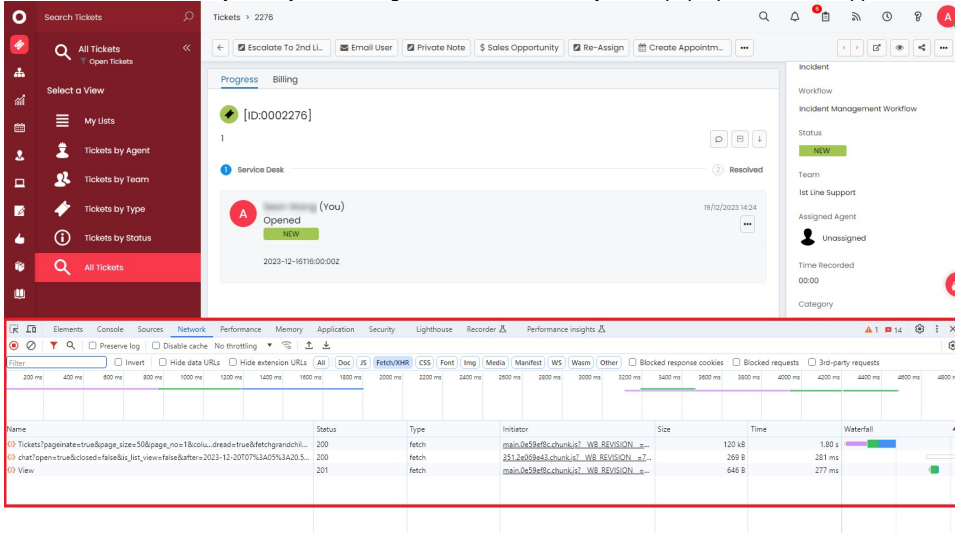
1. Log in to [Halo](#) app.



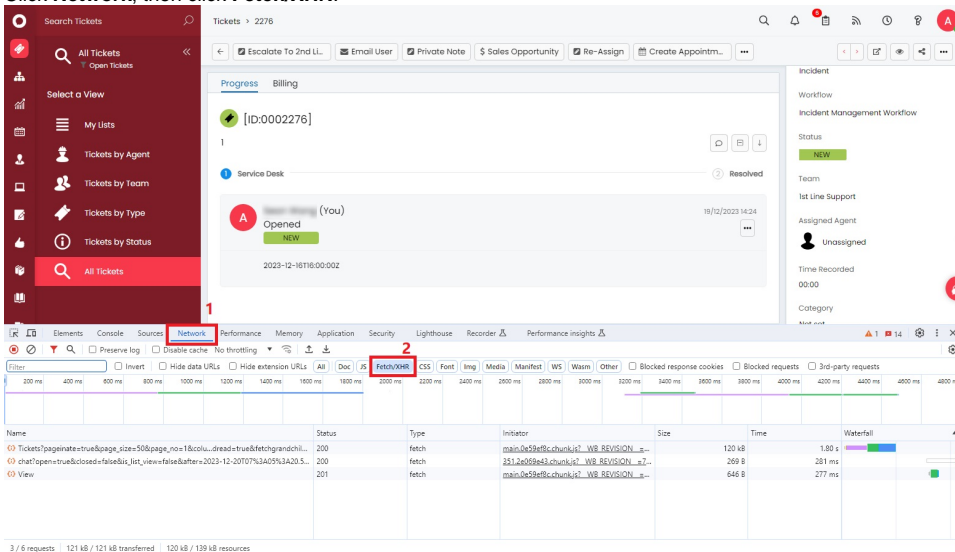
2. Open a ticket in the Halo app.



3. Press the **F12** button on your keyboard or right-click and select **Inspect**. A pop-up window will appear at the bottom of the page.



4. Click **Network**, then click **Fetch/XHR**.



5. On the ticket page in the Halo app, modify the **Status**, for example, change it from *NEW* to *IN PROGRESS*.

The screenshot shows the Halo app interface for a ticket with ID 2276. The status is currently 'NEW'. The interface includes a sidebar with navigation options, a top bar with filters, and a main content area showing ticket details and a list of requests.

Name	Status	Type	Initiator	Size	Time	Waterfall
Tickets?pageinate=true&page_size=50&page_no=1&column_dread=true&fetchgrndch...	200	fetch	main:de59ef63chuck:q2...WB REVISION...	120 kB	1.86 s	
View	201	fetch	main:de59ef63chuck:q2...WB REVISION...	647 B	282 ms	
chat?open=true&closed=false&id=2276-12-2070613459%3A205...	200	fetch	351:2e59ef63chuck:q2...WB REVISION...Z...	269 B	269 ms	
chat?open=true&closed=false&id=2276-12-2070613459%3A207...	200	fetch	351:2e59ef63chuck:q2...WB REVISION...Z...	269 B	281 ms	
View	201	fetch	main:de59ef63chuck:q2...WB REVISION...	648 B	1.31 s	
chat?open=true&closed=false&id=2276-12-2070613459%3A405...	200	fetch	351:2e59ef63chuck:q2...WB REVISION...Z...	269 B	283 ms	

6 / 10 requests 122 kB / 122 kB transferred 120 kB / 130 kB resources

6. In the window at the bottom of the page, click **Tickets**, then click **Payload**, and you will be able to see the status_id as 2 with IN PROGRESS.

The screenshot shows the Halo app interface for a ticket with ID 2276. The status is currently 'IN PROGRESS'. The interface includes a sidebar with navigation options, a top bar with filters, and a main content area showing ticket details and a list of requests. The network inspector at the bottom shows the 'Payload' tab for the 'Tickets' request, displaying the status_id as 2.

```
{
  "id": "2276",
  "files": null,
  "status_id": "2",
  "apply_rules": true,
  "utc_offset": -480,
  "refresh_response": true
}
```

Related Topics

- [What Filter Conditions and Formats are Available for Creating Widgets](#)
- [How to change the URL used by bots to send alerts](#)
- [Halo Integration Setup](#)
- [NextTicket Manager for Halo](#)
- [API Data Synchronization Failure or Pending in Halo Integration](#)
- [How to Create a Target Card Widget](#)
- [Creating a Scorecard Dashboard](#)
- [How to Create Slicers in Widgets](#)
- [Creating a Column Chart Widget](#)
- [Creating a Grid Widget](#)
- [Halo Public Datasets](#)
- [Creating the Line Chart and Stacked Line Widgets](#)
- [Bots: Types, Functions, and FAQs](#)
- [Bots](#)
- [MSPbots 3.0 Bots for ConnectWise Manage](#)