Adding ticket attachments with the API

ON THIS PAGE

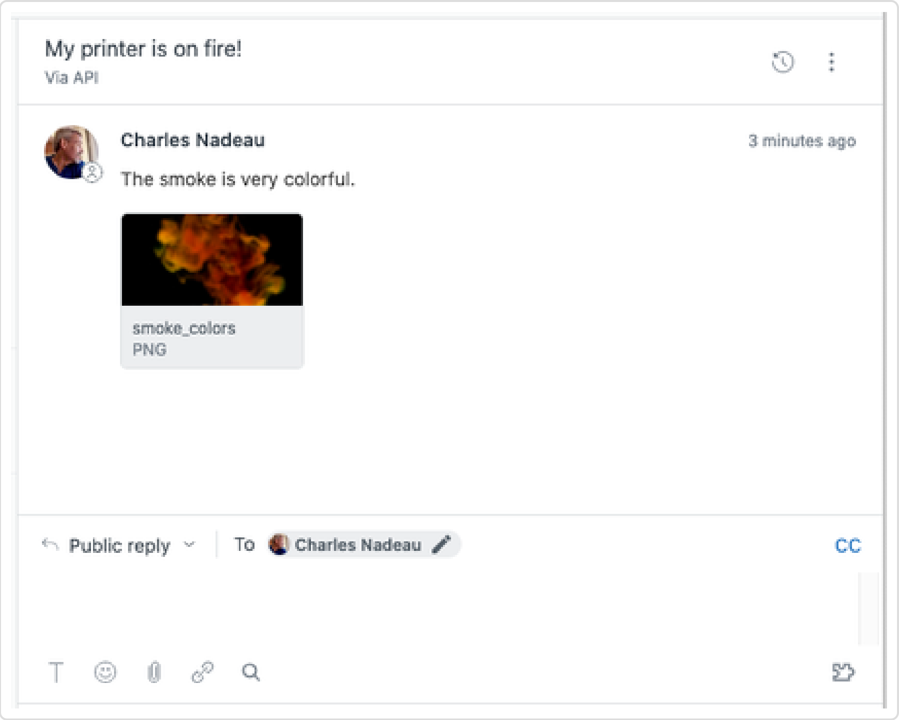
You can use the Zendesk API to attach files to tickets. The attachments are stored in Zendesk and added as links in ticket comments.

Before you start, make sure ticket attachments are enabled for end users in your account. See Allowing end users to attach files to tickets. If you're not a Zendesk administrator, ask one to check the setting for you.

Disclaimer: Zendesk provides this article for instructional purposes only. Zendesk does not support or guarantee the code. Zendesk also can't provide support for third-party technologies such as Postman or Python.

About ticket attachments

An attachment appears as a link in a ticket comment in the agent interface in Zendesk. Example:



My printer is on fire!
Via API

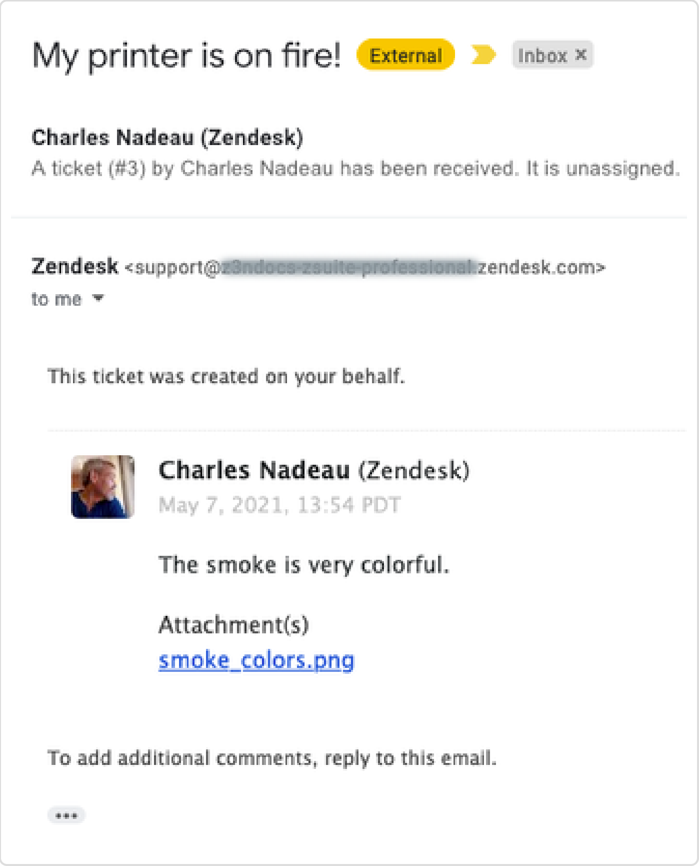
Charles Nadeau 3 minutes ago

The smoke is very colorful.

smoke_colors
PNG

You can preview image attachments in Zendesk. You can download other file types before opening them.

If ticket notifications are enabled, the attachment appears as a link in the notification email. Example:



My printer is on fire! External Inbox ×

Charles Nadeau (Zendesk)
A ticket (#3) by Charles Nadeau has been received. It is unassigned.

Zendesk <support@[redacted]zendesk.com>
to me

This ticket was created on your behalf.

Charles Nadeau (Zendesk)
May 7, 2021, 13:54 PDT

The smoke is very colorful.

Attachment(s)
smoke_colors.png

To add additional comments, reply to this email.

By default, attachments are secured using a URL with an attachment token that is considerably complex and random. However, the attachments are visible to anyone with the URL and token. If an email notification is misdirected to someone other than the intended end user, sensitive information may be accidentally exposed. You can enable private attachments. See Enabling private attachments in the Zendesk help center.

Ticket attachments work as follows in Zendesk:

- You can only attach files to comments in tickets, not to the tickets themselves.
- You can only attach files when adding the comments to tickets. You can’t attach files to existing comments in tickets.
- You can only add comments with attached files to tickets when creating or updating tickets using the Tickets API, not the Ticket Comments API, which doesn't support adding comments to tickets.
- You cannot remove a comment from a ticket. However, you can redact an attachment from a comment. See Redact Comment Attachment.
- You upload the files you want to attach separately using the Upload Files endpoint in the Attachments API.
- You can attach any type of file. The file size limit is 50 MB.

Adding a ticket attachment

Adding an attachment to a ticket with the API consists of two steps:

1. Uploading the file you want to attach
2. Attaching the file to a new ticket comment

Step 1 - Uploading the file you want to attach

Use the Upload Files endpoint in the Attachments API to upload the file you want to attach to a ticket comment:

```
POST /api/v2/uploads?filename={filename}
```

The request only uploads the file; it doesn't attach it to a ticket. However, the request will return a token that you can use in step 2 to attach the file to a ticket.

Example request:

```
curl "https://example.zendesk.com/api/v2/uploads.json?filename=order_issue.png" \
  --data-binary @screenshot_02.png \
  -H "Content-Type: image/png" \
  -v -u {email_address}/token:{api_token} \
  -X POST
```

The `filename` query parameter is required. It determines what the file will be named when attached to the ticket. It doesn't specify the file on your system that you want to upload. In the example, the local file to be uploaded is named **screenshot_02.png** but it will be renamed **order_issue.png** in the ticket. Changing the name helps give the agent more context about the attachment.

While the two names can be different, their file extensions must match. If they don't match, the agent's browser or file reader could give an error when it tries to open the attachment. From the application's standpoint, the content of the file will be incompatible with the file extension.

The `Content-Type` header must contain a recognized MIME type that correctly describes the type of the uploaded file. Failing to send a recognized, correct type may cause undesired behavior. For example, in-browser audio playback may be interrupted by the browser's security mechanisms for MP3s uploaded with an incorrect type.

Because you can't specify an **application/json** content-type header for this request, you must append **.json** to the endpoint url:

```
.../api/v2/uploads.json...
```

If the file was uploaded successfully by the request, the response returns a `token` value, among other data. Save the token. You'll use it to attach the file to a ticket in the next step. The token is valid until its `expires_at` time.

Example response:

```
{
  "upload": {
    "token": "4bLLKSOU63CPqaIeOMXYyXzUh",
    "expires_at": "2021-05-08T22:50:18Z",
    "attachment": {
      "url": "https://example.zendesk.com/api/v2/attachments/1503194928902.json",
      "id":1503194928902,
      "file_name": "order_issue.png",
      "content_url": "https://example.zendesk.com/attachments/token/vp7DnuiSvehLZtK2yrPjqJ1l6/?name=order_issue.png",
      "content_type": "image/png"
    },
    "attachments": [
      {
        "url": "https://example.zendesk.com/api/v2/attachments/1503194928902.json",
        "id":1503194928902,
        "file_name": "order_issue.png",
        "content_url": "https://example.zendesk.com/attachments/token/vp7DnuiSvehLZtK2yrPjqJ1l6/?name=order_issue.png"
,
        "content_type": "image/png"
      }
    ]
  }
}
```

Step 2 - Attaching the file to a new ticket comment

After saving the `token` from the upload, you can use it to attach the file to a ticket comment.

You cannot attach the file directly to a ticket. You can only attach the file to a ticket comment, and then only when adding the comment to a ticket. You can't attach a file to an existing ticket comment.

You can only add a comment to a ticket by creating or updating the ticket with the Tickets API.

When creating or updating a ticket with a new comment, specify one or more upload tokens in the `uploads` property of your `comment` object. The following Update Ticket example adds a comment that includes the file uploaded in the first step:

```
curl https://example.zendesk.com/api/v2/tickets/45135 \
  -d '{"ticket": {"comment": {"body": "Press play", "uploads": ["4bLLKSOU63CPqaIeOMXYyXzUh"]}}}' \
  -H "Content-Type: application/json" \
  -v -u {email_address}/token:{api_token} -X PUT
```

The token is single-use. After you use a token to attach a file to a ticket comment, you can't use the same token to attach the same file to another ticket comment.

You can verify that the file was attached to the comment by viewing the ticket in the agent interface or by using the List Comments endpoint.

Examples

This section shows how to add ticket attachments using curl, Postman, and Python:

- curl example
- Postman example
- Python example

Each fictional example attaches the same image to the first comment of a new ticket. The request details of each example are as follows:

- Local file name: **img_0603.png**
- Attachment file name: **smoke_colors.png**
- Account name: example.zendesk.com
- User email: jdoe@example.com
- API token: xpfRUSj51oGQta26nhRyaLaIBdhf

Replace these values when testing attachments. If successful, the attachment will look like the examples in About tickets attachments.

curl example

To upload the file

1. In your command line interface (CLI), navigate to the folder that contains the **img_0603.png** image.
2. Run the following command in the CLI:

```
curl "https://example.zendesk.com/api/v2/uploads.json?filename=smoke_colors.png" \
  --data-binary @img_0603.png \
  -H "Content-Type: image/png" \
  -v -u jdoe@example.com/token:xpfRUSj51oGQta26nhRyaLaIBdhf \
  -X POST
```

   Use the `--data-binary` option to specify the file to upload with the request. The `@` flag points to a file on your system.

   In the example, the file is in the same directory as the directory where you run the curl request. If the file is in a folder other than the one you use to run the curl request, specify the relative path. Example:

   `--data-binary @images/img_0603.png`

   If the file name contains spaces or other characters that might confuse curl, enclose the name in quotes. If the file name doesn't have a file extension, make sure to include one in the curl command. For example, if the file on your system is named **Screen Shot 2021-05-06 at 9.41.08 AM**, then use:

   `--data-binary @"Screen Shot 2021-05-06 at 9.41.08 AM.png"`

   The `Content-Type` header must contain a recognized MIME type that correctly describes the type of the uploaded file. Failing to send a recognized, correct type may cause undesired behavior. For example, in-browser audio playback may be interrupted by the browser's security mechanisms for MP3s uploaded with an incorrect type.
3. If the request is successful, retrieve the `token` value from the response and save it.

   Assume the request returned the following `token` property:

`"token":"hX0TsMPY8sDHrpmnUtDFqqNtJ"`.

To add the file to the ticket

1. Run the following command to create a ticket with the file attached to the first comment:

   ```
   curl https://example.zendesk.com/api/v2/tickets \
     -d '{"ticket": {"subject": "My printer is on fire!", "comment": {"body": "The smoke is very colorful.", "uploads": ["hX0TsMPY8sDHrpmnUtDFqqNtJ"]}}}' \
     -H "Content-Type: application/json" \
     -v -u jdoe@example.com/token:xpfRUSj51oGQta26nhRyaLaIBdhf -X POST
   ```

   The `uploads` property specifies the upload token.
2. Look for the new ticket in Zendesk and verify the file is attached to the first comment.

Postman example

To upload the file

1. In Postman, create a request by selecting **File** > **New** and selecting **Request**.

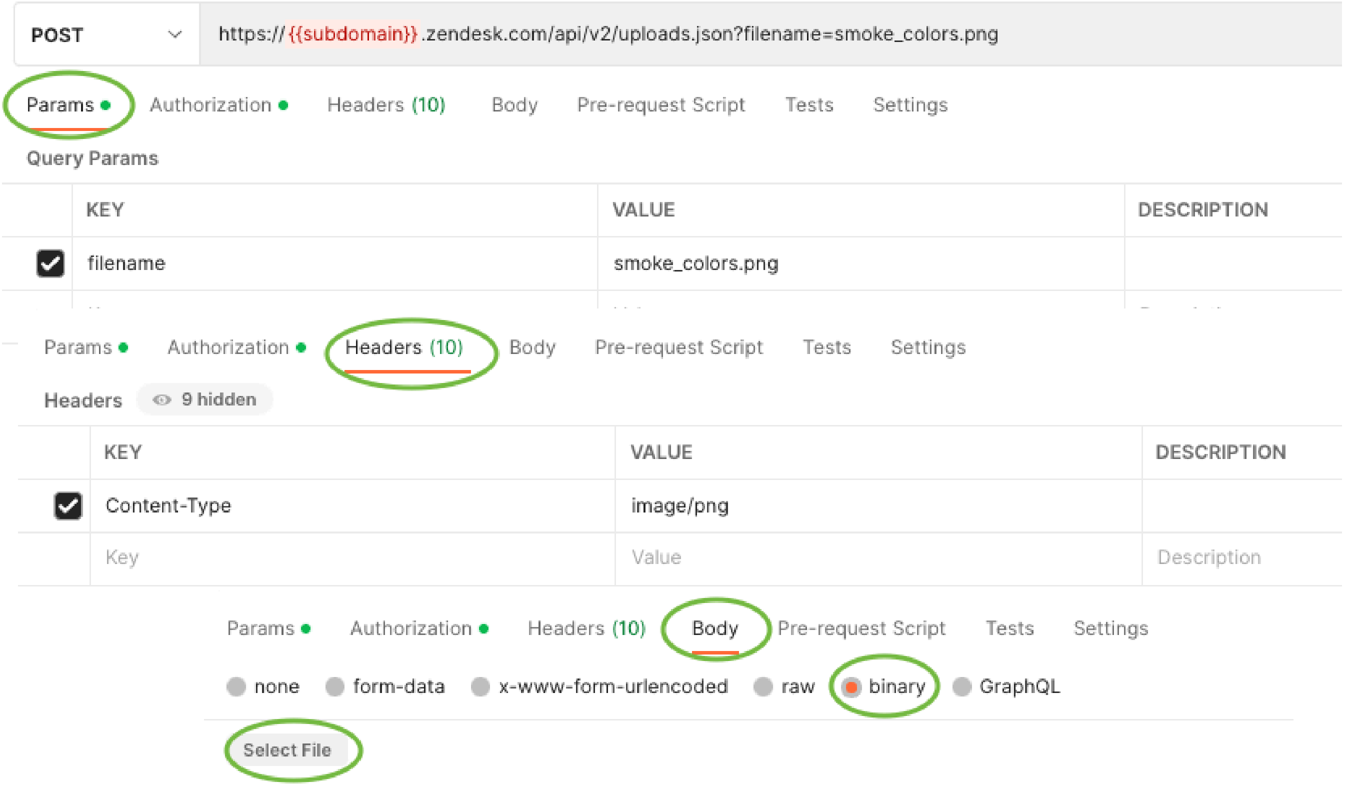
   Name the request **Upload Files**, assign it to a collection, and click **Save**.
2. In the request path, select **POST** and enter the following path:

   `https://example.zendesk.com/api/v2/uploads.json`
3. In the **Params** tab, add a key named **filename** with a value of **smoke_colors.png**.
4. In the **Authorization** tab, select **Basic Auth** from the menu, and enter the following settings:

   - Username: jdoe@example.com/token
   - Password: xpfRUSj51oGQta26nhRyaLaIBdhf

   You can also set the credentials at the collection level and then select **Inherit auth from parent** from the menu.
5. In the **Headers** tab, add a key named **Content-Type** with a value of **image/png**.
6. In the **Body** tab, select the **binary** radio button. Click Select Files and select the **img_0603.png** file.

   Postman should look like this, but with a different `filename` value:



POST https://{{subdomain}}.zendesk.com/api/v2/uploads.json?filename=smoke_colors.png

Query Params

	KEY	VALUE	DESCRIPTION
☑	filename	smoke_colors.png	

	KEY	VALUE	DESCRIPTION
☑	Content-Type	image/png	
	Key	Value	Description

Body: ● binary

7. Click **Send** to make the request.
8. If the request is successful, retrieve the `token` value from the response and save it.

   Assume the request returned the following `token` property:

   `"token":"hX0TsMPY8sDHrpmnUtDFqqNtJ"`.

To add the file to the ticket

1. In Postman, create a new request named **Create Ticket**.
2. For the request path, select **POST** and enter the following path:

   `https://example.zendesk.com/api/v2/tickets.json`
3. In the **Authorization** tab, use the following credentials:

   - Username: jdoe@example.com/token
   - Password: xpfRUSj51oGQta26nhRyaLalBdhf

4. In the **Body** tab, select the **raw** radio button, select **JSON** from the menu, and paste the following payload into the panel:

```
{
  "ticket": {
    "subject": "My printer is on fire!",
    "comment": {
      "body": "The smoke is very  colorful.",
      "uploads": [
        "hX0TsMPY8sDHrpmnUtDFqqNtJ"
      ]
    }
  }
}
```

The `uploads` property specifies the upload token.

5. Click **Send** to make the request.
6. Look for the new ticket in Zendesk and verify the file is attached to the first comment.

Python example

The following Python script uploads a file and attaches it to a comment in a new ticket. It uses the popular requests library to simplify the request.

```python
import requests


local_filename = 'img_0603.png'
attachment_filename = 'smoke_colors.png'

email = 'jdoe@example.com'
api_token = 'xpfRUSj51oGQta26nhRyaLaIBdhf'
auth = (f'{email}/token', api_token)

# upload file
url = 'https://example.zendesk.com/api/v2/uploads.json'
params = {'filename': attachment_filename}
headers = {'Content-Type': 'image/png'}
with open(local_filename, 'rb') as f:
    response = requests.post(url, params=params, data=f, headers=headers, auth=auth).json()
upload_token = response['upload']['token']

# attach file
url = 'https://example.zendesk.com/api/v2/tickets'
payload = {
    'ticket': {
        'subject': 'My printer is on fire!',
        'comment': {
            'body': 'The smoke is very  colorful.',
            'uploads': [upload_token]
        }
    }
}
```

```
response = requests.post(url, json=payload, auth=auth).json()
print(response)
```

How it works

For this simple example, the script starts by assigning values to the `local_filename` and `attachment_filename` variables. You'd normally get these values from function arguments or by prompting the user. Example:

```
local_filename = input('Enter the name of the local file: ')
attachment_filename = input('Enter a name for the attachment: ')
```

Next, the script defines an `auth` request parameter to authorize the request:

```
email = 'jdoe@example.com'
api_token = 'xpfRUSj51oGQta26nhRyaLaIBdhf'
auth = (f'{email}/token', api_token)
```

The script also defines `params` and `headers` request parameters, which are required for attachments:

```
params = {'filename': attachment_filename}
headers = {'Content-Type': 'image/png'}
```

The script opens the **local_filename** file in binary mode and passes the file-like object to the request (along with the previously defined `params`, `headers`, and `auth` parameters):

```
with open(local_filename, 'rb') as f:
    response = requests.post(url, params=params, data=f, headers=headers, auth=auth).json()
```

The response object's `json()` method converts the response's JSON data into Python data.

The script retrieves the `token` value from the response data:

```
upload_token = response['upload']['token']
```

The rest of the script creates a ticket containing a comment, which includes the upload token to attach the file.

Related topics