# API guide: How to locate unknown users in your data

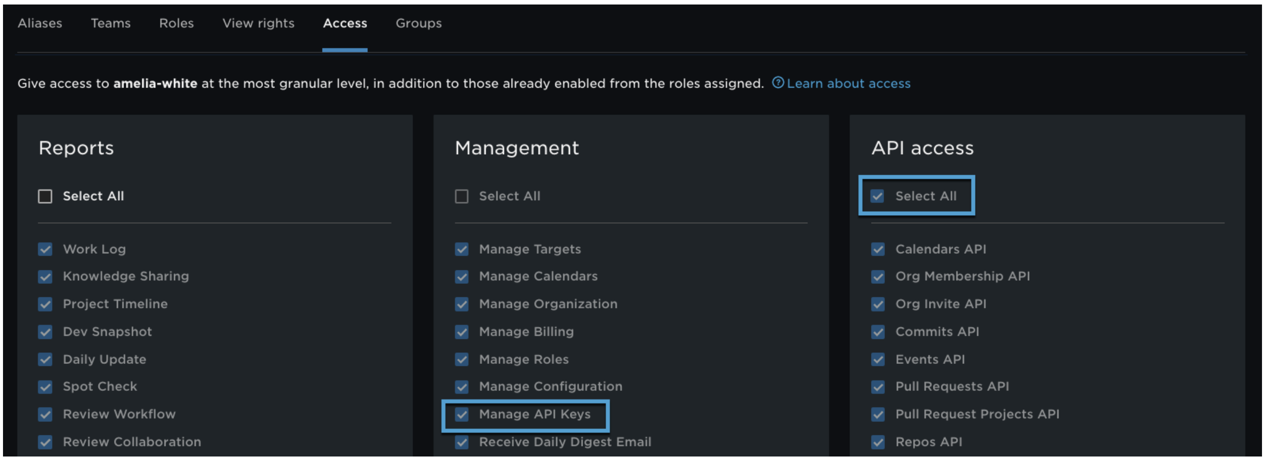
Flow ingests any user that has been associated with an imported repo or project. This could come in the form of a commit, comment, ticket subscription etc. This usually means there are users in your account that you do not recognize. This document will walk you through how to locate where a user was ingested from, using the API.

In this article

## Permissions

First, let's make sure you have access to the API and all its endpoints. You will need Manage API keys and API access permissions.

On the user page, select the Access tab. Under Management, select **Manage API Keys**. Under API access, select **Select All**.



Give access to **amelia-white** at the most granular level, in addition to those already enabled from the roles assigned. ⓘ Learn about access

Reports

- ☐ Select All
- ☑ Work Log
- ☑ Knowledge Sharing
- ☑ Project Timeline
- ☑ Dev Snapshot
- ☑ Daily Update
- ☑ Spot Check
- ☑ Review Workflow
- ☑ Review Collaboration

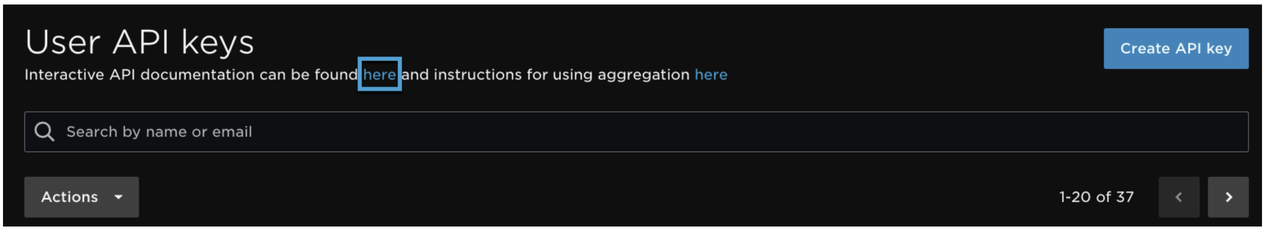
Management

- ☐ Select All
- ☑ Manage Targets
- ☑ Manage Calendars
- ☑ Manage Organization
- ☑ Manage Billing
- ☑ Manage Roles
- ☑ Manage Configuration
- ☑ Manage API Keys
- ☑ Receive Daily Digest Email

API access

- ☑ Select All
- ☑ Calendars API
- ☑ Org Membership API
- ☑ Org Invite API
- ☑ Commits API
- ☑ Events API
- ☑ Pull Requests API
- ☑ Pull Request Projects API
- ☑ Repos API

With your permissions set, navigate to **Settings,** then click **API keys** and click on the link "Interactive API documentation can be found here.



User API keys

Interactive API documentation can be found here and instructions for using aggregation here

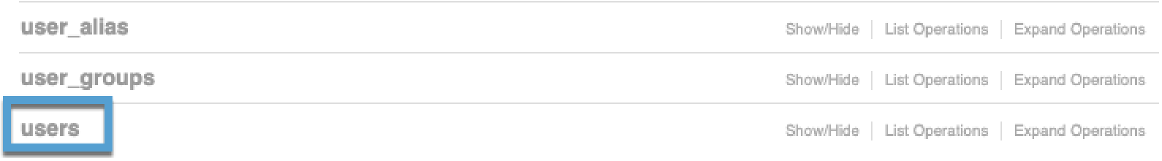
A new tab will open and you will see all your API endpoints.

back to top

## Locate the user ID

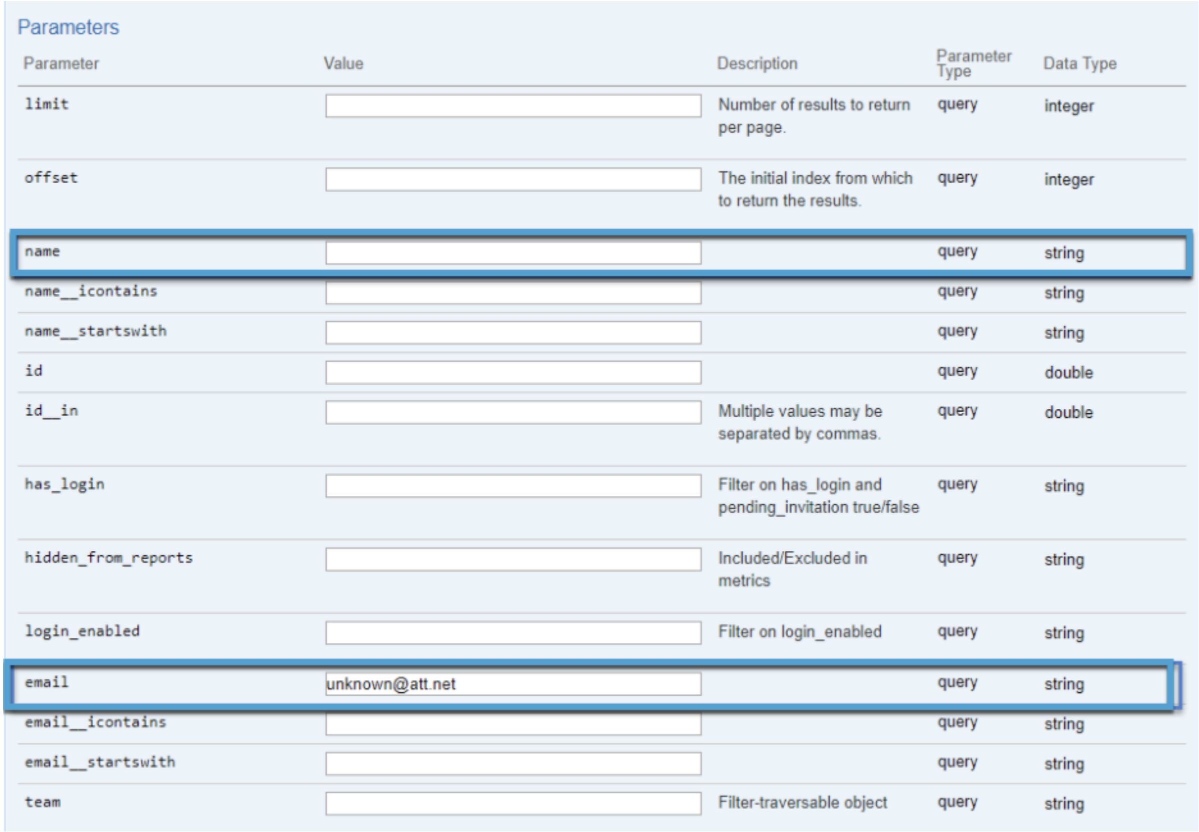
First we need to identify the unknown user's user_id so we can use that to locate any work he/she is associated with.

1. Click on the very last endpoint, **users**.



| | | | |
|---|---|---|---|
| user_alias | Show/Hide | List Operations | Expand Operations |
| user_groups | Show/Hide | List Operations | Expand Operations |
| users | Show/Hide | List Operations | Expand Operations |

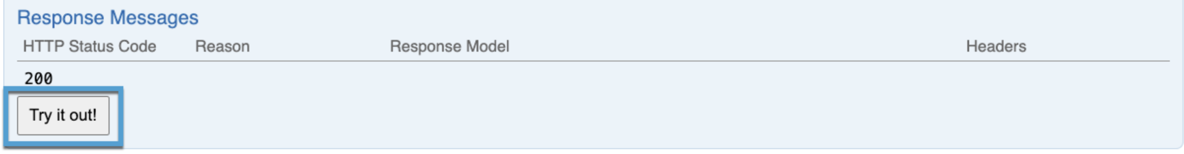
2. A list of available parameters will appear. You can locate the user_id with the name or email of the user. Type in the name or email of the unknown user in the appropriate field. In this example we are using the email address.



Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| limit | | Number of results to return per page. | query | integer |
| offset | | The initial index from which to return the results. | query | integer |
| name | | | query | string |
| name__icontains | | | query | string |
| name__startswith | | | query | string |
| id | | | query | double |
| id__in | | Multiple values may be separated by commas. | query | double |
| has_login | | Filter on has_login and pending_invitation true/false | query | string |
| hidden_from_reports | | Included/Excluded in metrics | query | string |
| login_enabled | | Filter on login_enabled | query | string |
| email | unknown@att.net | | query | string |
| email__icontains | | | query | string |
| email__startswith | | | query | string |
| team | | Filter-traversable object | query | string |

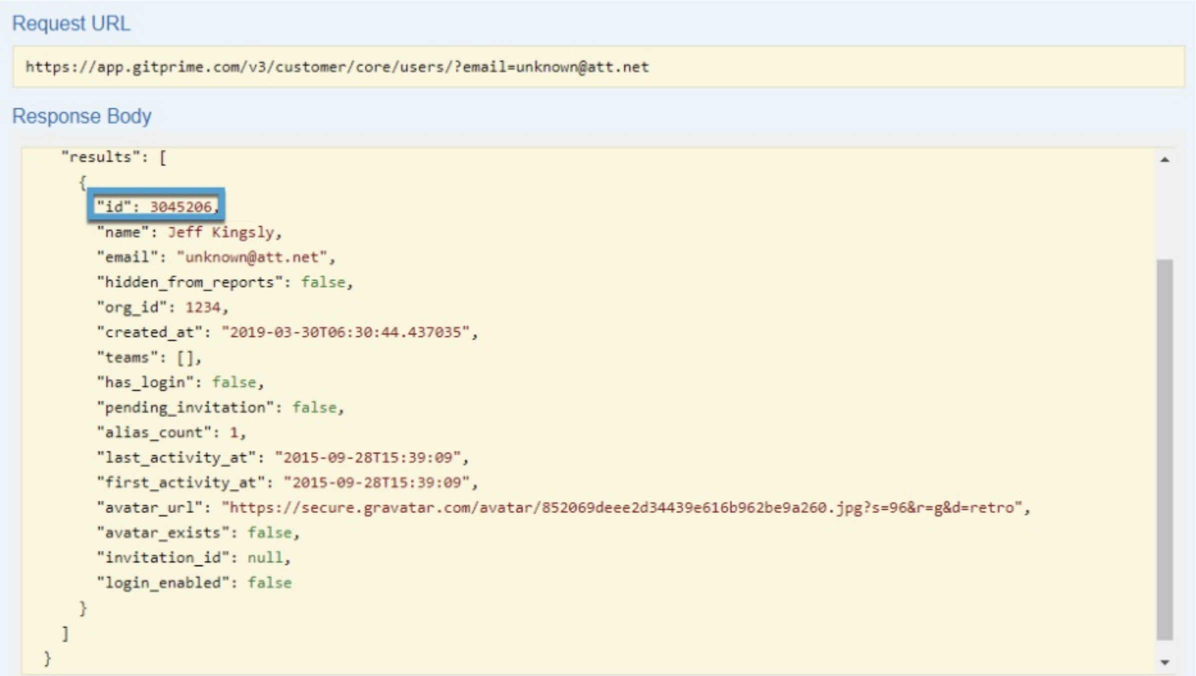
3. Hit "Enter" or click the **Try it out**button at the bottom of the parameter list.



Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|---|---|---|---|
| 200 | | | |

4. If you have typed in the name and/or email correctly you will see a result set like this for the unknown user. The user_id we need will be listed at the top of the results next to the parameter "id". Copy this number to your clipboard. We will be using this in the following steps.



Request URL

```
https://app.gitprime.com/v3/customer/core/users/?email=unknown@att.net
```

Response Body

```
  "results": [
    {
      "id": 3045206,
      "name": Jeff Kingsly,
      "email": "unknown@att.net",
      "hidden_from_reports": false,
      "org_id": 1234,
      "created_at": "2019-03-30T06:30:44.437035",
      "teams": [],
      "has_login": false,
      "pending_invitation": false,
      "alias_count": 1,
      "last_activity_at": "2015-09-28T15:39:09",
      "first_activity_at": "2015-09-28T15:39:09",
      "avatar_url": "https://secure.gravatar.com/avatar/852069deee2d34439e616b962be9a260.jpg?s=96&r=g&d=retro",
      "avatar_exists": false,
      "invitation_id": null,
      "login_enabled": false
    }
  ]
}
```

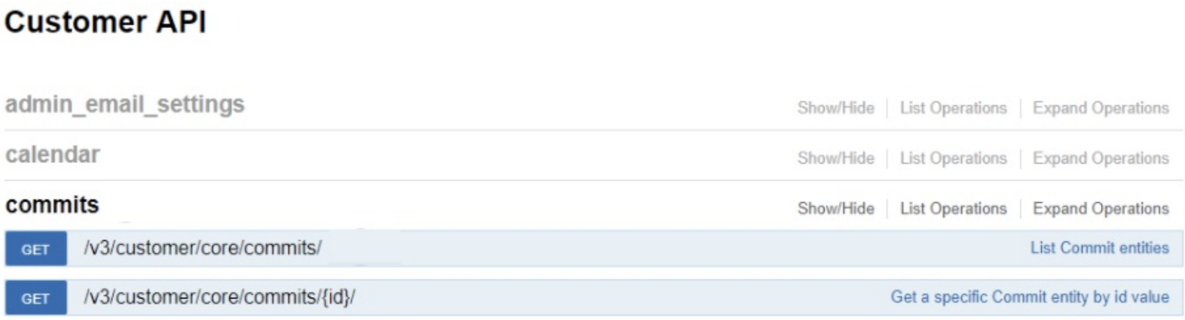
Now that you have the user_id we can locate any work that has been imported for this user.

back to top

# Commits

Let's check to see if the unknown user has any commits recorded across any of your imported repos.

1. Scroll to the top of your API endpoint list and click on **commits**,then click the top **GET** URL. This will open up the list of parameters for the commits endpoint.



Customer API

admin_email_settings

Show/Hide | List Operations | Expand Operations

calendar

Show/Hide | List Operations | Expand Operations

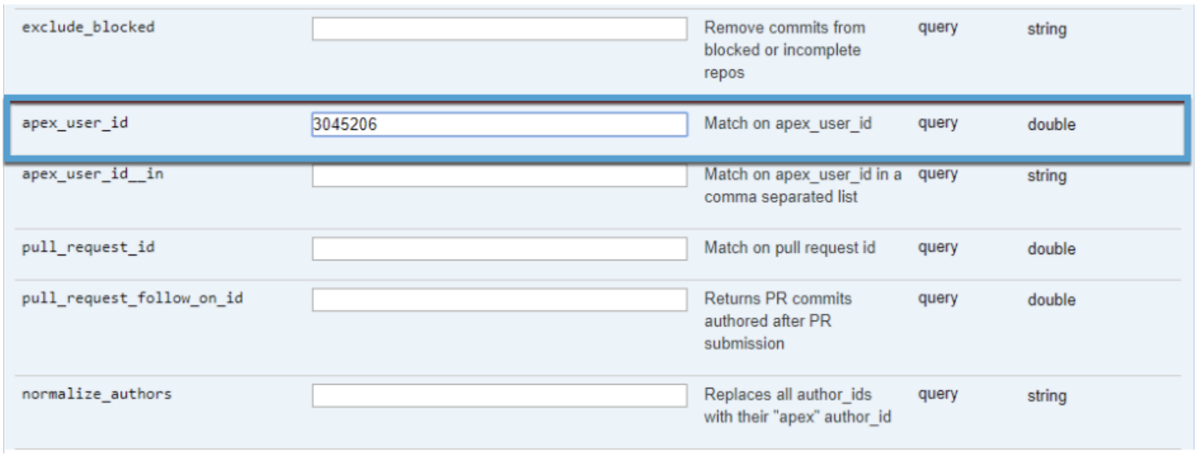
commits

Show/Hide | List Operations | Expand Operations

GET /v3/customer/core/commits/ List Commit entities

GET /v3/customer/core/commits/{id}/ Get a specific Commit entity by id value

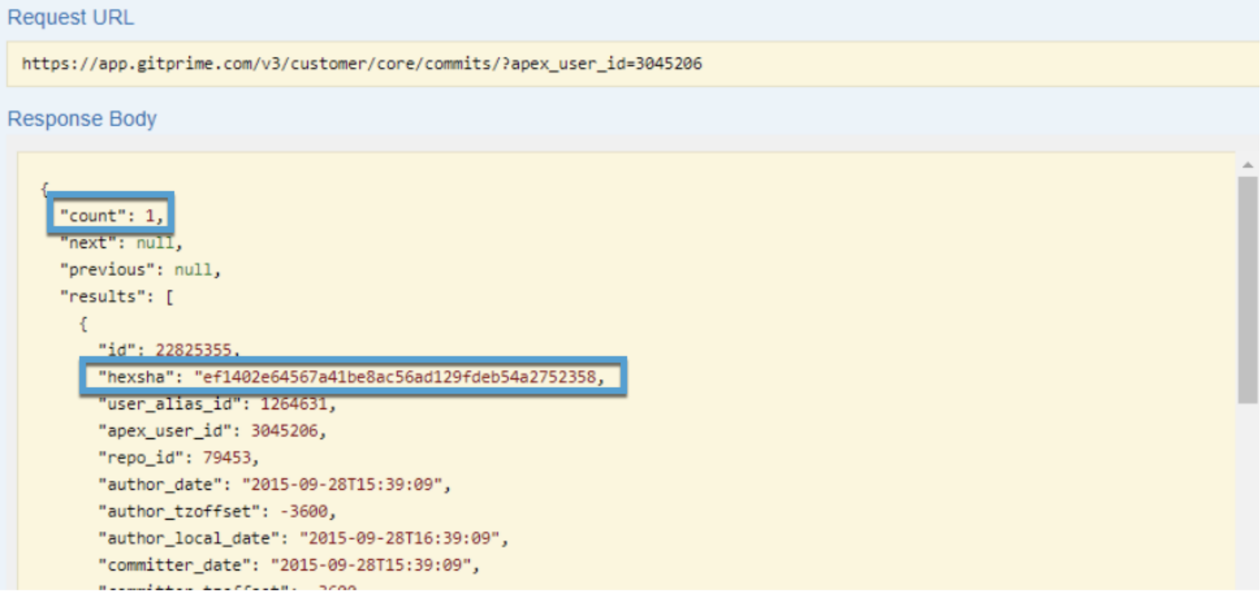
2. Locate the parameter **apex_user_id.**It will be towards the bottom of your parameter list. Paste in the user_id you copied in the last section and hit "Enter" or use the **Try it out** button to run the query.



| | | | | |
|---|---|---|---|---|
| exclude_blocked | | Remove commits from blocked or incomplete repos | query | string |
| apex_user_id | 3045206 | Match on apex_user_id | query | double |
| apex_user_id__in | | Match on apex_user_id in a comma separated list | query | string |
| pull_request_id | | Match on pull request id | query | double |
| pull_request_follow_on_id | | Returns PR commits authored after PR submission | query | double |
| normalize_authors | | Replaces all author_ids with their "apex" author_id | query | string |

3. If any results appear you can identify what commits the unknown User authored by locating the "external_commit_url". This will take you into your Git host to review the work further. If there are commits for this User the results will look like this.

Success! You found the source of your unknown user. Count is the total number of records/commits found. Hexsha is the commit sha from Git host.



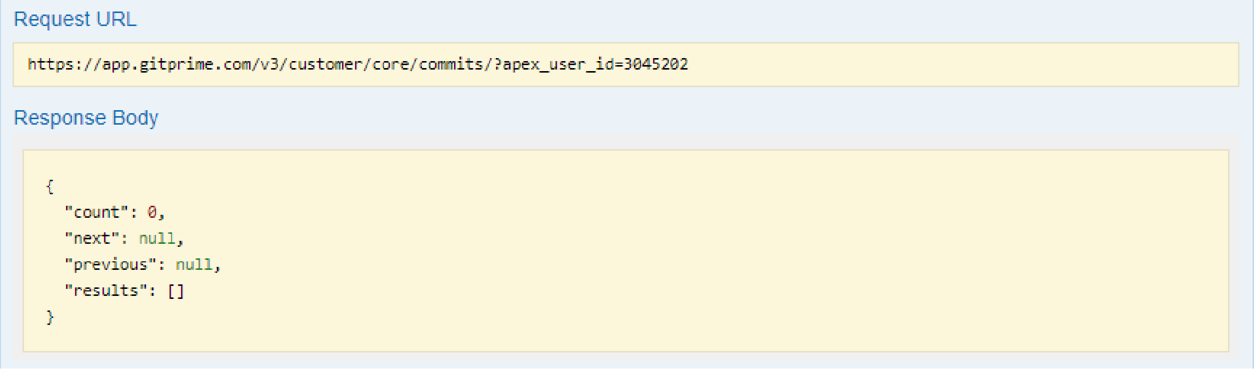
Request URL

```
https://app.gitprime.com/v3/customer/core/commits/?apex_user_id=3045206
```

Response Body

```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 22825355,
      "hexsha": "ef1402e64567a41be8ac56ad129fdeb54a2752358",
      "user_alias_id": 1264631,
      "apex_user_id": 3045206,
      "repo_id": 79453,
      "author_date": "2015-09-28T15:39:09",
      "author_tzoffset": -3600,
      "author_local_date": "2015-09-28T16:39:09",
      "committer_date": "2015-09-28T15:39:09",
      "committer_tzoffset": -3600
```

If there were 0 commits found for this user your response will look like this. Move on to the next section **Pull requests** to continue looking.



Request URL

```
https://app.gitprime.com/v3/customer/core/commits/?apex_user_id=3045202
```

Response Body

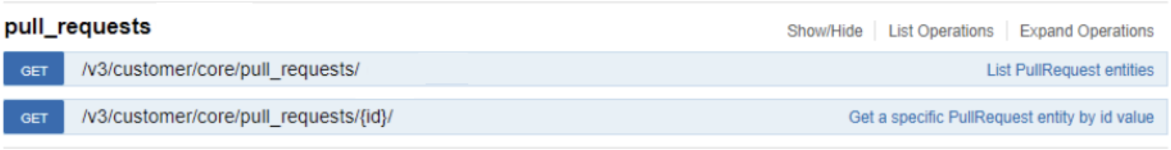
```
{
  "count": 0,
  "next": null,
  "previous": null,
  "results": []
}
```

back to top

# Pull requests

If the commit's endpoint did not return any results. Let's continue the search in the **pull request**endpoint.

1. Locate and click on the **pull request** endpoint.

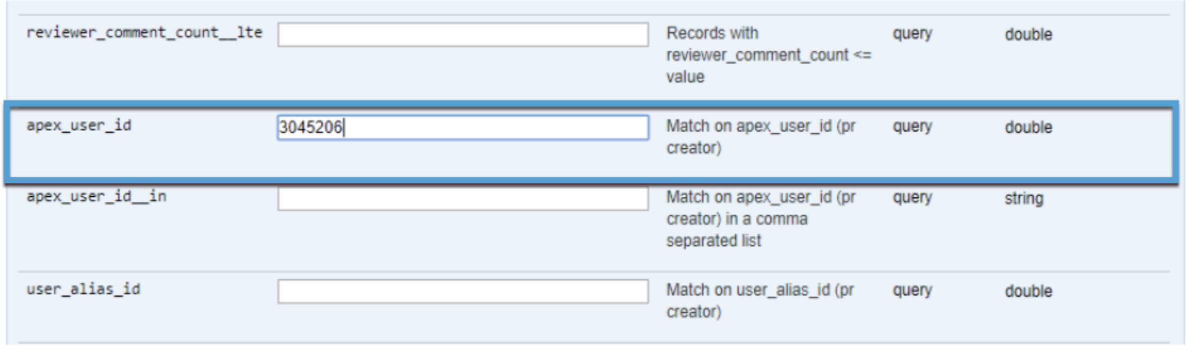


**pull_requests**

GET /v3/customer/core/pull_requests/ — List PullRequest entities

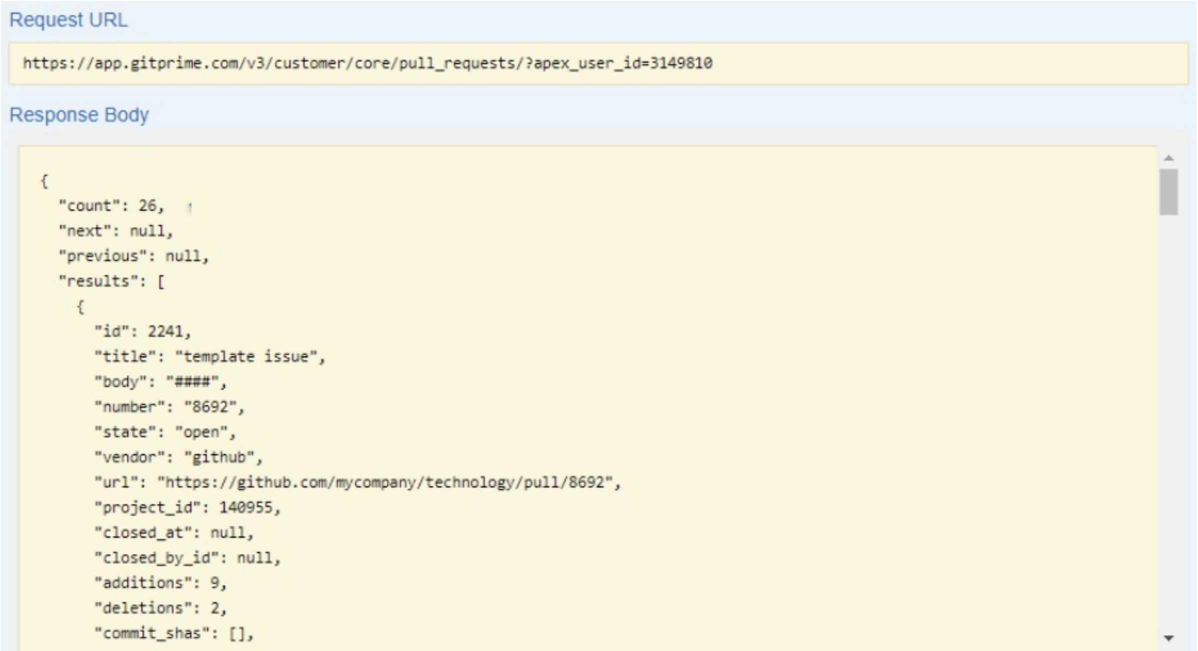
GET /v3/customer/core/pull_requests/{id}/ — Get a specific PullRequest entity by id value

2. Find the **apex_user_id**parameter and paste in your user_id and hit "Enter" or use the **Try it out** button.



| | | | | |
|---|---|---|---|---|
| reviewer_comment_count__lte | | Records with reviewer_comment_count <= value | query | double |
| apex_user_id | 3045206 | Match on apex_user_id (pr creator) | query | double |
| apex_user_id__in | | Match on apex_user_id (pr creator) in a comma separated list | query | string |
| user_alias_id | | Match on user_alias_id (pr creator) | query | double |

3. Upon running your query you will either see results or you will not. If 0 PRs were found continue to the next endpoint. If results were found for this unknown user you have identified the origin source for the user.



Request URL

```
https://app.gitprime.com/v3/customer/core/pull_requests/?apex_user_id=3149810
```

Response Body

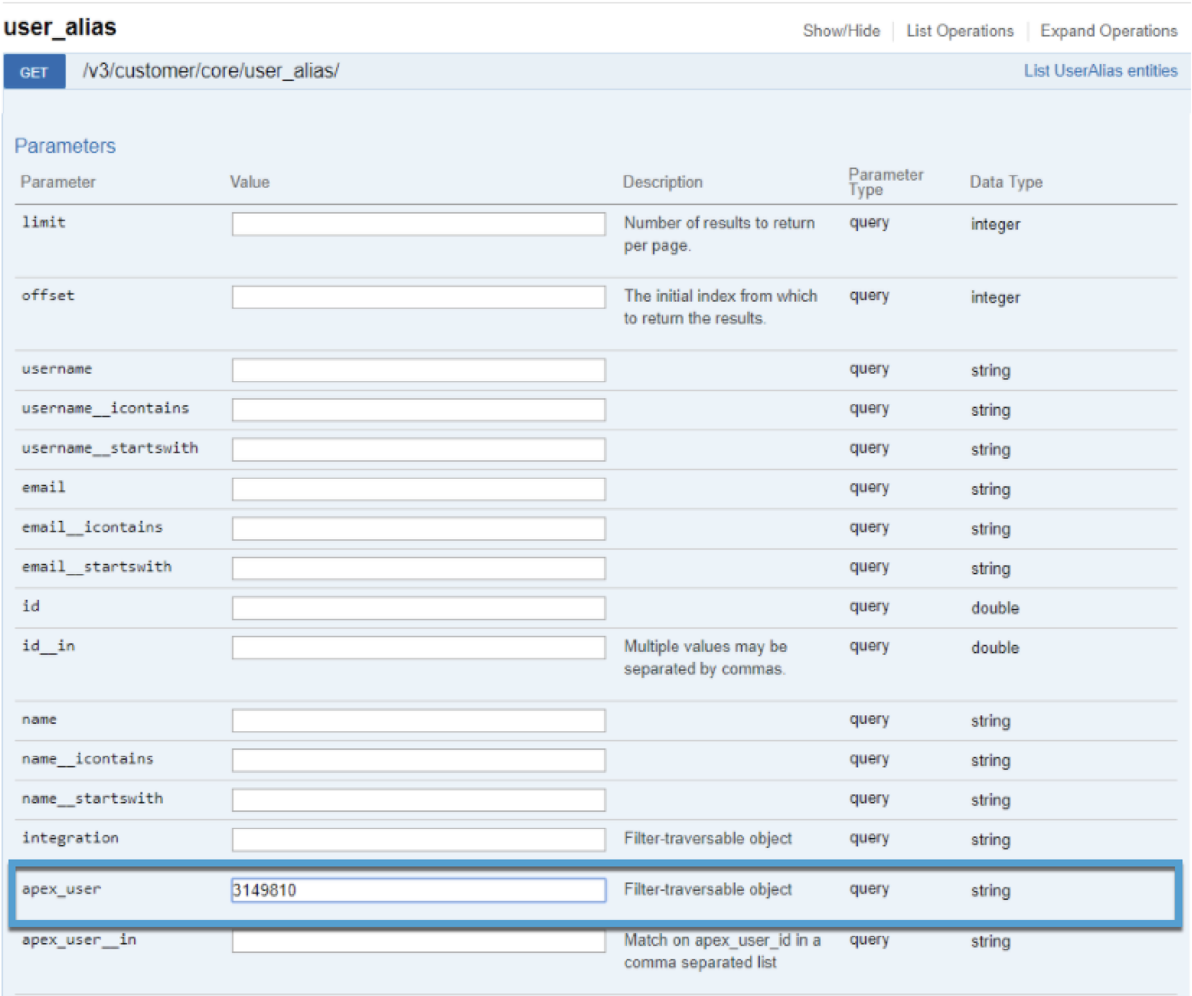
```
{
  "count": 26,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 2241,
      "title": "template issue",
      "body": "####",
      "number": "8692",
      "state": "open",
      "vendor": "github",
      "url": "https://github.com/mycompany/technology/pull/8692",
      "project_id": 140955,
      "closed_at": null,
      "closed_by_id": null,
      "additions": 9,
      "deletions": 2,
      "commit_shas": [],
```

back to top

# Tickets

If you have a ticket integration or use your Git host tickets/issues you can look for your unknown user in the tickets endpoints. Ticket endpoints require an user_alias_id. These are the aliases that have been merged under the apex_user. You should try the apex_user_id and any user_alias_ids to locate tickets, ticket events and/or ticket comments this unknown User may have made.

1. Under the **user_alias**endpoint locate the **apex_user**field and paste the user_id you have been using in the previous steps. Hit "Enter" or click the **Try it out** button.



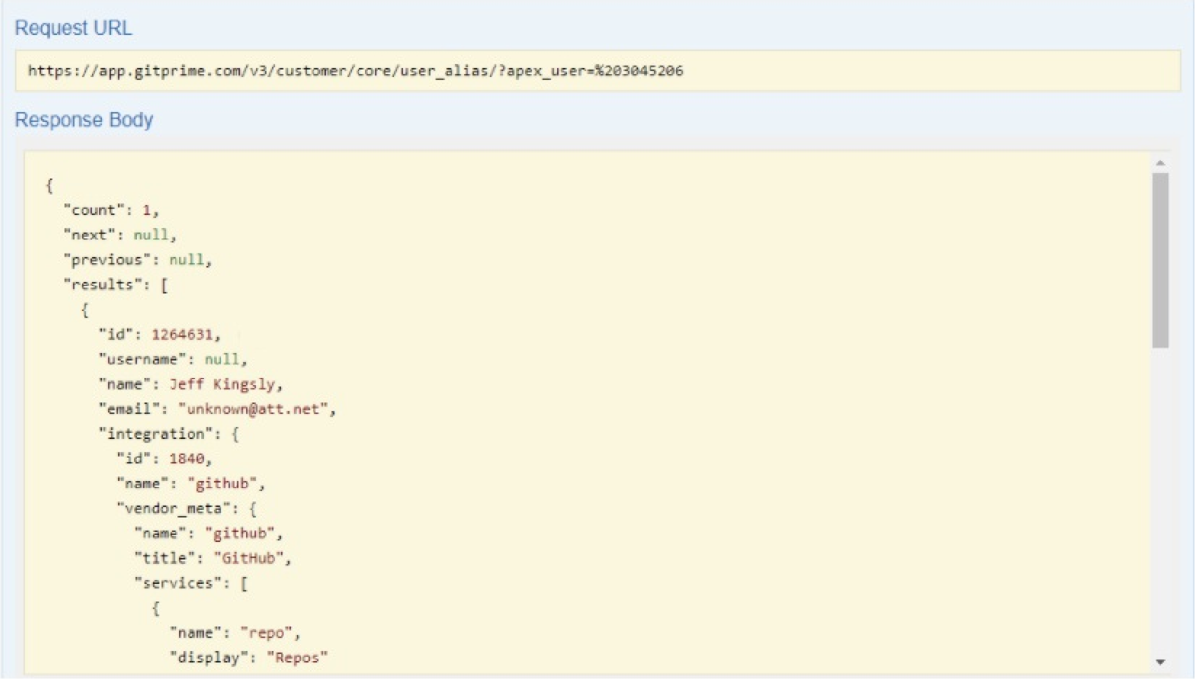
user_alias

/v3/customer/core/user_alias/

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| limit | | Number of results to return per page. | query | integer |
| offset | | The initial index from which to return the results. | query | integer |
| username | | | query | string |
| username__icontains | | | query | string |
| username__startswith | | | query | string |
| email | | | query | string |
| email__icontains | | | query | string |
| email__startswith | | | query | string |
| id | | | query | double |
| id__in | | Multiple values may be separated by commas. | query | double |
| name | | | query | string |
| name__icontains | | | query | string |
| name__startswith | | | query | string |
| integration | | Filter-traversable object | query | string |
| apex_user | 3149810 | Filter-traversable object | query | string |
| apex_user__in | | Match on apex_user_id in a comma separated list | query | string |

2. The results will show you all the aliases found for this apex_user. Count is the total number of aliases found for the user. Copy and save all id's for later use. Id is user_alias_id.



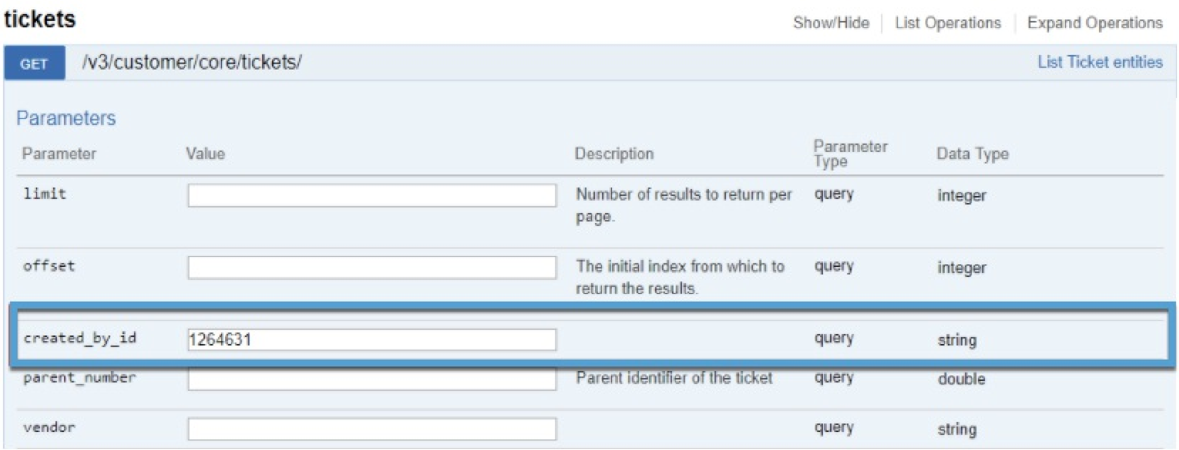
Request URL

```
https://app.gitprime.com/v3/customer/core/user_alias/?apex_user=%203045206
```

Response Body

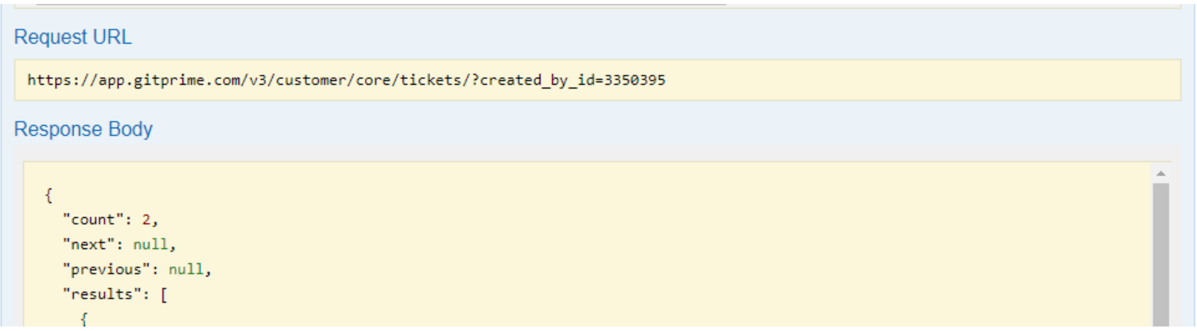
```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 1264631,
      "username": null,
      "name": Jeff Kingsly,
      "email": "unknown@att.net",
      "integration": {
        "id": 1840,
        "name": "github",
        "vendor_meta": {
          "name": "github",
          "title": "GitHub",
          "services": [
            {
              "name": "repo",
              "display": "Repos"
```

3. Now that we have all user_alias_ids for our unknown user let's see if we can locate any tickets. Open the **tickets** endpoint and locate the **created_by_id** field



| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| limit | | Number of results to return per page. | query | integer |
| offset | | The initial index from which to return the results. | query | integer |
| created_by_id | 1264631 | | query | string |
| parent_number | | Parent identifier of the ticket | query | double |
| vendor | | | query | string |

4. Copy one of the user_alias_ids you saved from **Step: 2** and hit "Enter" or **Try it out** to see if any records are found. Swap out each user_alias_id and try the apex_user_id (the original ID we got at the beginning of this guide). If records are found, great! If not continue on to the next endpoint.



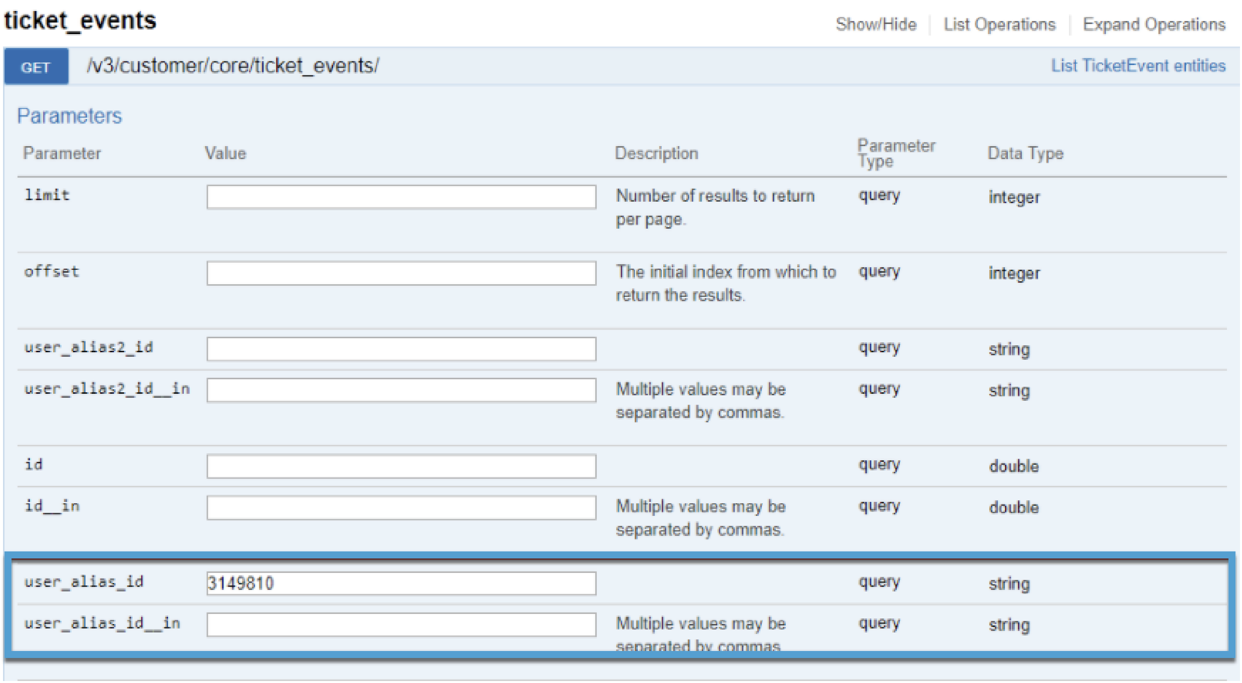
```
https://app.gitprime.com/v3/customer/core/tickets/?created_by_id=3350395
```

```
{
  "count": 2,
  "next": null,
  "previous": null,
  "results": [
    {
```

back to top

## Ticket events

Locate and click on the **ticket_events**endpoint. Find the **user_alias_id** field and paste in one of your user_alias_ids. You can also utilize the user_alias_id__in field and paste in all your user_alias_id's separated by a comma. This will look for any record across any of the ID's you input. Click **Try it out** to see the results.



**ticket_events**

/v3/customer/core/ticket_events/

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| limit | | Number of results to return per page. | query | integer |
| offset | | The initial index from which to return the results. | query | integer |
| user_alias2_id | | | query | string |
| user_alias2_id__in | | Multiple values may be separated by commas. | query | string |
| id | | | query | double |
| id__in | | Multiple values may be separated by commas. | query | double |
| user_alias_id | 3149810 | | query | string |
| user_alias_id__in | | Multiple values may be separated by commas. | query | string |

back to top

# Next endpoints

If you have been unsuccessful in locating any data in the above endpoints, continue searching across the remaining endpoints listed below.

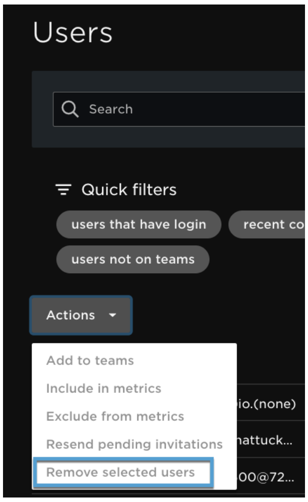
- ticket_comments
- pull_request_events
- pull_request_comments

back to top

# I did not find any data for the user

I did not find any data for the user

If you have not find any data in your account for this user, it could be that the data has been deleted/removed from your account. This can happen if the repo or project has been removed from your Flow account. If this the case, you should be able to delete this user as there is no data tying them to your account.You can do this on the **User** page.



back to top

If you need help, please email support@pluralsight.com () for 24/7 assistance.