

Obtaining and using access tokens

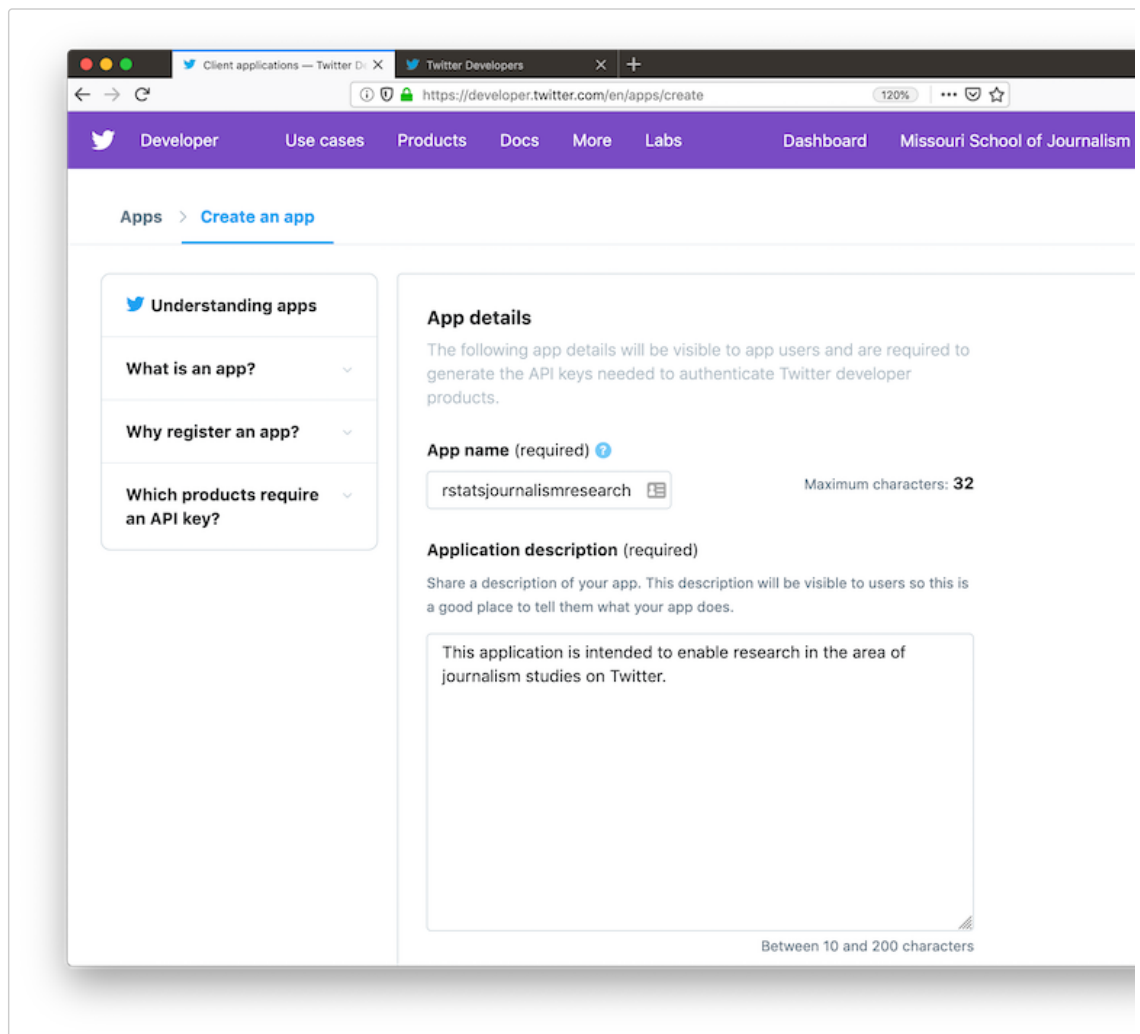
rtweet: Collecting Twitter Data

rtweet

This vignette covers how to obtain and use Twitter API access tokens for use in the `rtweet` package.

Creating a Twitter App

- To create a Twitter app, you will first need to apply for a developer account. This process along with a detailed explanation can be found at [developer.twitter.com](https://developer.twitter.com/en/apps).
- Once you have acquired a developer account, navigate to developer.twitter.com/en/apps, click the blue button that says, Create a New App, and then complete the form with the following fields:
 - App Name: What your app will be called
 - Application Description: How your app will be described to its users



The screenshot shows the 'Create an app' page on the Twitter Developer portal. The page has a purple header with navigation links: Developer, Use cases, Products, Docs, More, Labs, Dashboard, and Missouri School of Journalism. Below the header, the breadcrumb 'Apps > Create an app' is visible. On the left, there is a sidebar titled 'Understanding apps' with three expandable sections: 'What is an app?', 'Why register an app?', and 'Which products require an API key?'. The main content area is titled 'App details' and contains the following information:

- App details**: The following app details will be visible to app users and are required to generate the API keys needed to authenticate Twitter developer products.
- App name (required)**: A text input field containing 'rstatsjournalismresearch' with a character count of 32. A note indicates 'Maximum characters: 32'.
- Application description (required)**: A text area containing the description: 'This application is intended to enable research in the area of journalism studies on Twitter.' A note at the bottom indicates 'Between 10 and 200 characters'.

- Website URLs: Website associated with app—I recommend using the URL to your Twitter profile
- Callback URLs: **IMPORTANT** enter exactly the following: `http://127.0.0.1:1410`

The screenshot shows the 'Create app' page on the Twitter Developer portal. The browser tabs include 'Client applications — Twitter D...' and 'Twitter Developers'. The address bar shows 'https://developer.twitter.com/en/apps/create'. The navigation bar has links for Developer, Use cases, Products, Docs, More, Labs, Dashboard, and Missouri School of Journalism. The form fields are as follows:

- Website URL (required)**: A text input field containing 'ps://twitter.com/kearneywm'.
- Allow this application to be used to sign in with Twitter**: A checkbox labeled 'Enable Sign in with Twitter' which is currently unchecked. A 'Learn more' link is present.
- Callback URLs**: A text input field containing 'http://127.0.0.1:1410/'. Below it is a '+ Add another' link.
- Terms of Service URL**: A text input field containing 'https://'.
- Privacy policy URL**: A text input field containing 'https://'.
- Organization name**: An empty text input field.
- Organization website URL**: A text input field containing 'https://'.

- Tell us how this app will be used: Be clear and honest

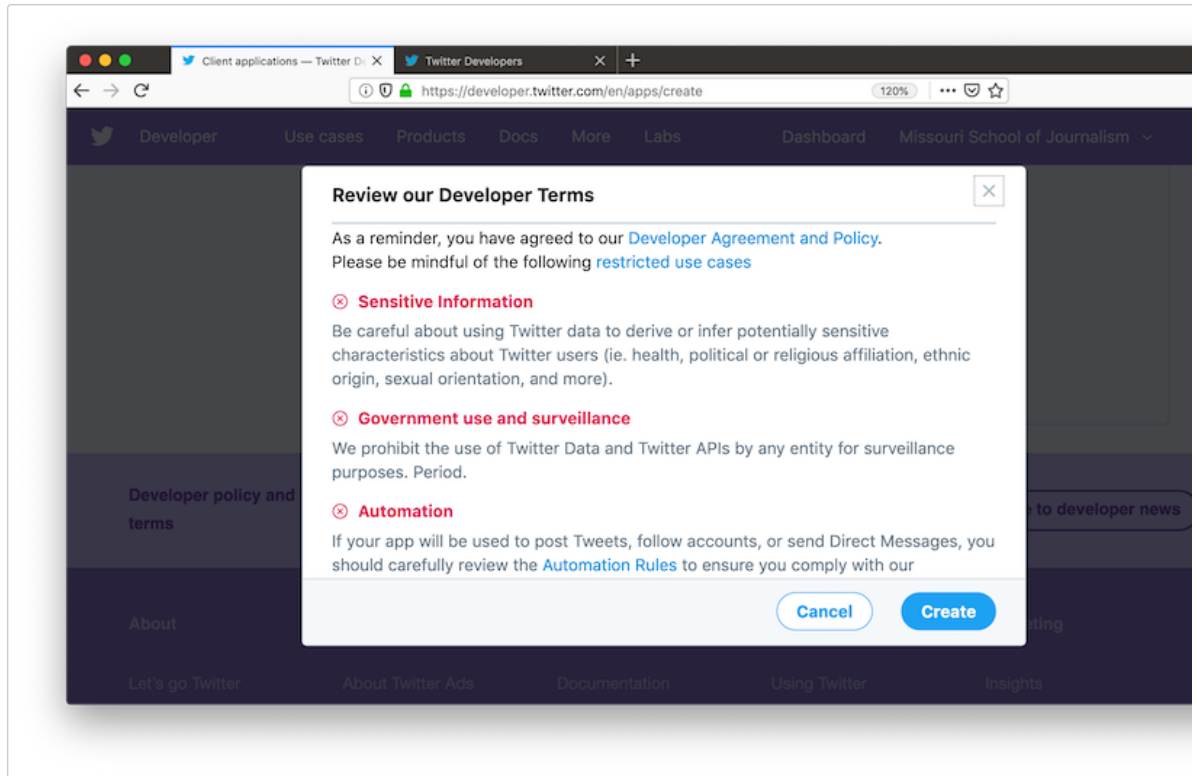
The screenshot shows the 'Tell us how this app will be used' section of the 'Create app' page. The text area contains the following description:

I plan to use this app for academic research focusing on journalism on Twitter—e.g., how journalists use Twitter, how news travels on Twitter, etc.

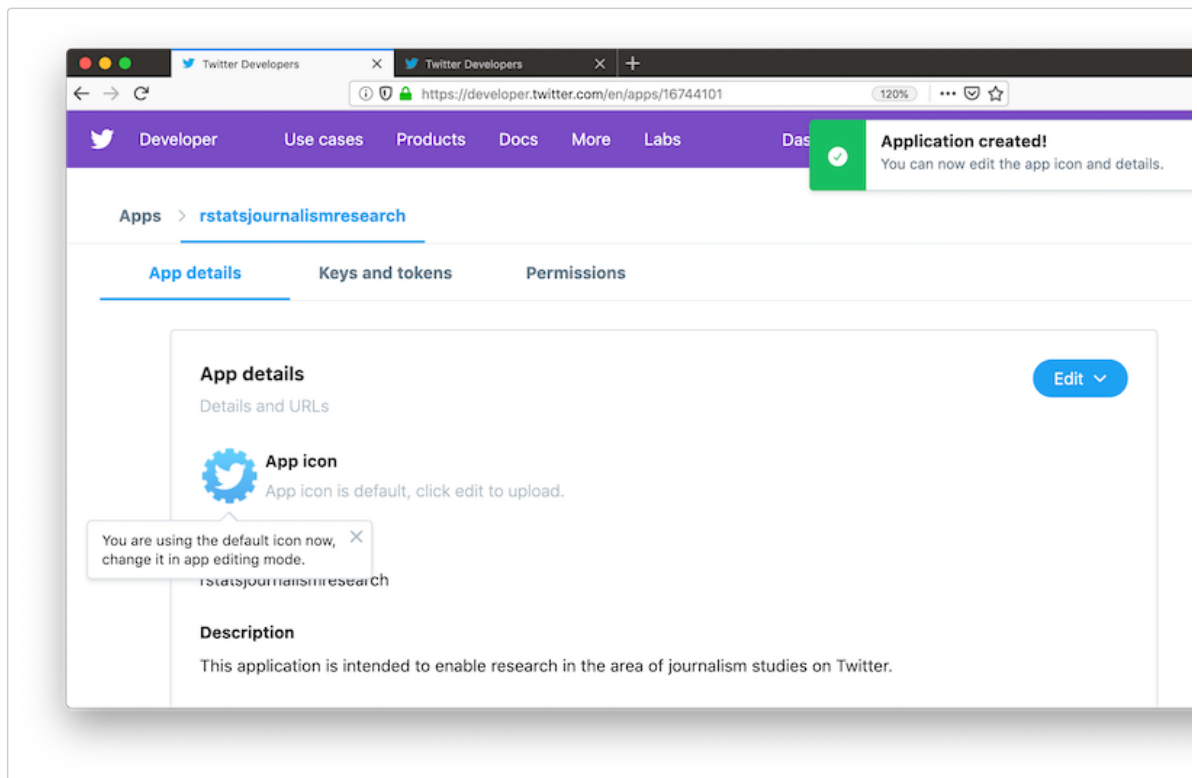
At the bottom of the form are two buttons: 'Cancel' and 'Create'.

- When you've completed the required form fields, click the blue Create button at the bottom

- Read through and indicate whether you accept the developer terms



- And you're done!



Authorization methods

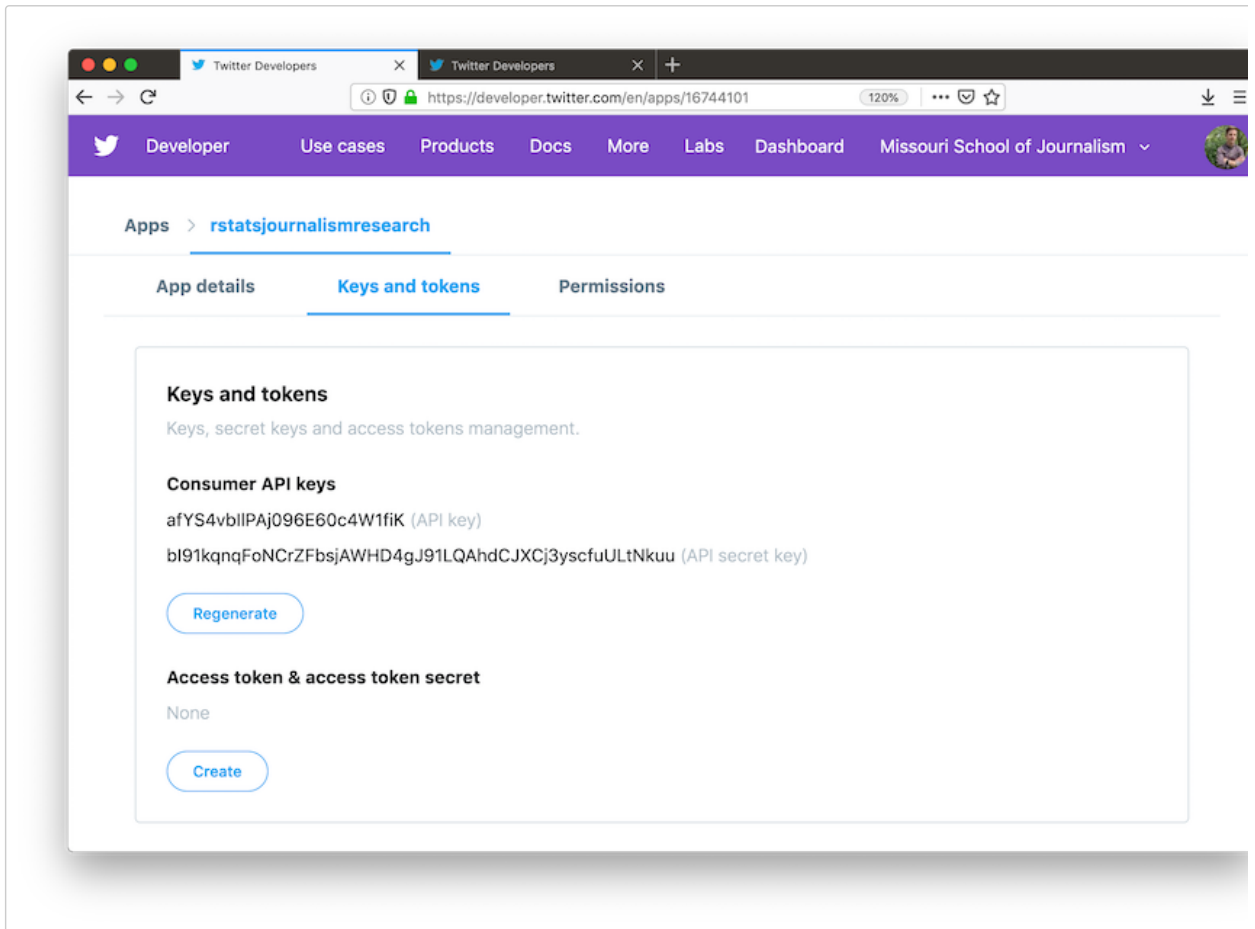
Users can create their personal Twitter token in two different ways. Each method is outlined below.

1. Browser-based authentication

- Authentication via web browser requires the `httpuv` package to be installed.

```
## install httpuv if not already
if (!requireNamespace("httpuv", quietly = TRUE)) {
  install.packages("httpuv")
}
```

- Navigate to developer.twitter.com/en/apps and select your Twitter app
- Click the tab labeled Keys and tokens to retrieve your keys.
- Locate the Consumer API keys (aka "API Secret").



- Copy and paste the two keys (along with the name of your app) into an R script file and pass them along to `create_token()`.

```
## load rtweet
library(rtweet)

## store api keys (these are fake example values; replace with your own keys)
api_key <- "afYS4vbIlPAj096E60c4W1fiK"
api_secret_key <- "bI91kqnqFoNCrZFbsjAWHD4gJ91LQAhdCJXCj3yscfuULtNkuu"

## authenticate via web browser
token <- create_token(
  app = "rstatsjournalismresearch",
  consumer_key = api_key,
  consumer_secret = api_secret_key)
```

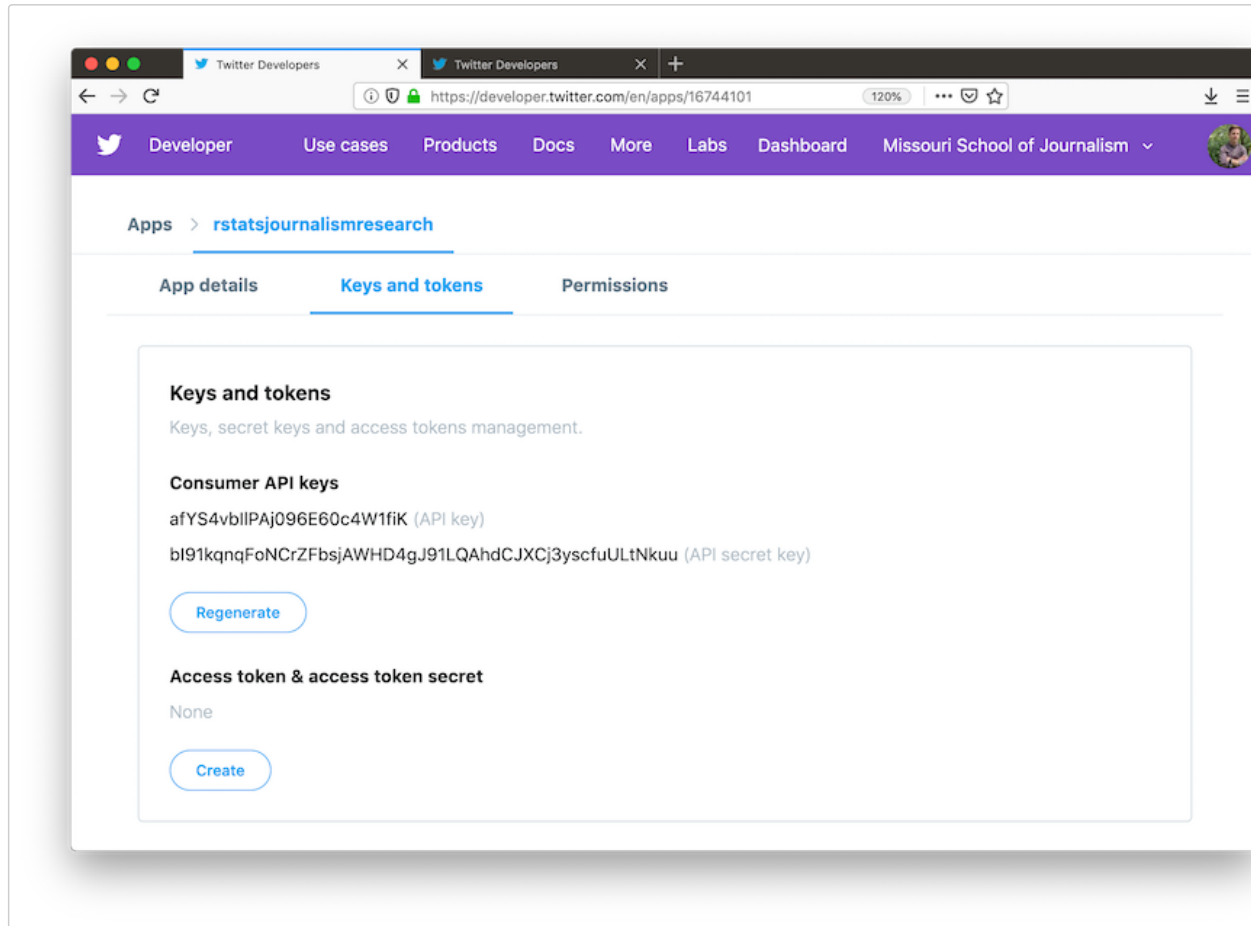
- A browser pop-up window should appear. Click to approve (must be signed into twitter.com) and return to R.
- Your token has been created. Print your token to make sure the app name and `api_key` match

```
## view token (you should see the correct app name)
token
```

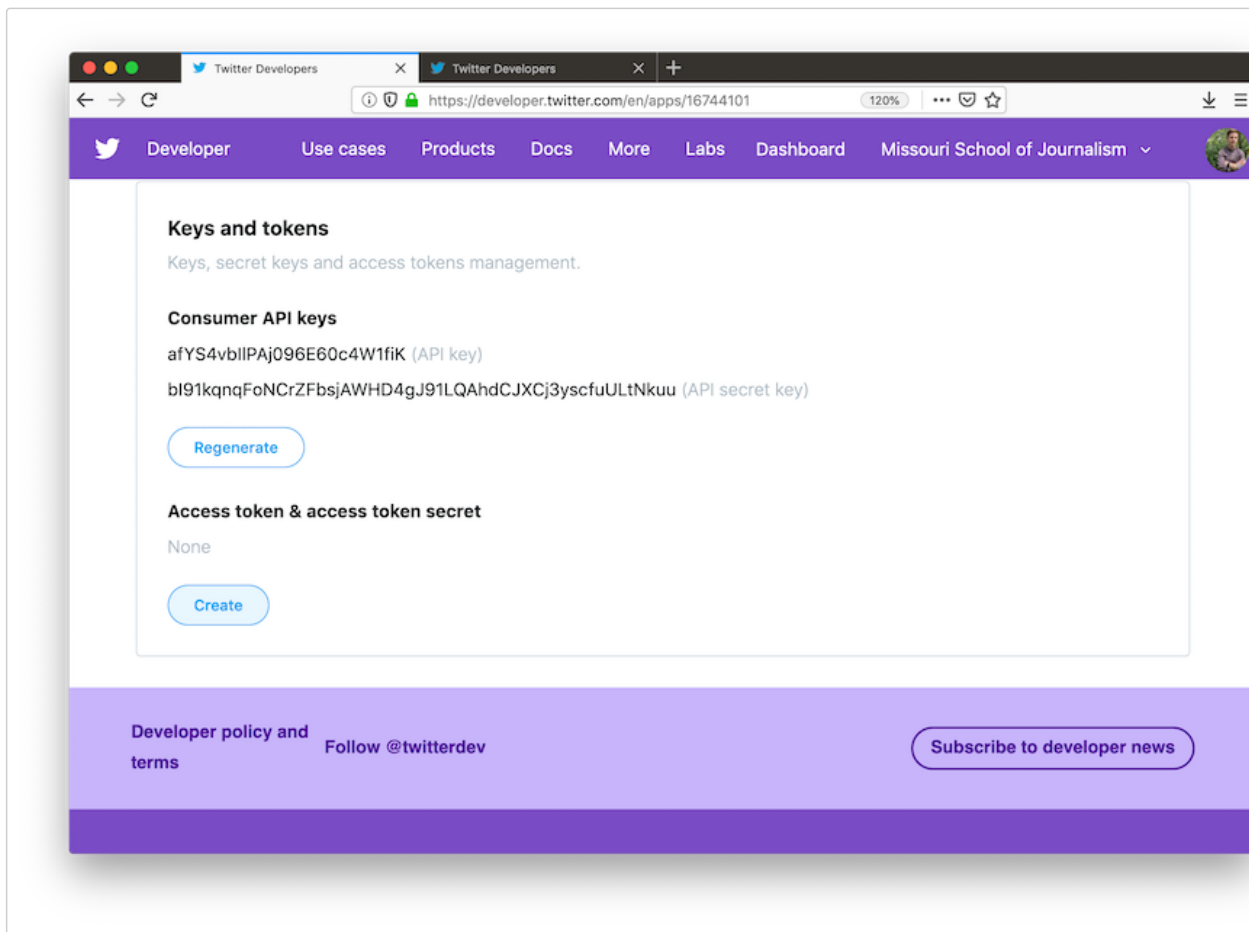
2. Access token/secret method

- Navigate to developer.twitter.com/en/apps and select your Twitter app
- Click the tab labeled Keys and tokens to retrieve your keys.

- Locate the Consumer API keys (aka “API Secret”).



- Scroll down to Access token & access token secret and click Create



- Copy and paste the four keys (along with the name of your app) into an R script file and pass them along to `create_token()`.

```
## store api keys (these are fake example values; replace with your own keys)
api_key <- "afYS4vbIlPAj096E60c4W1fiK"
api_secret_key <- "bl91kqnqFoNCrZFbsjAWHD4gJ91LQAhdCJXCj3yscfuULtNkuu"
access_token <- "9551451262-wK2EmA942kxZYIwa5LMKZoQA4Xc2uyIiEwu2YXL"
access_token_secret <- "9vpiSGKg1fIPQtxc5d5ESiFLZQpfbknEN1f1m2xe5byw7"

## authenticate via web browser
token <- create_token(
  app = "rstatsjournalismresearch",
  consumer_key = api_key,
  consumer_secret = api_secret_key,
  access_token = access_token,
  access_secret = access_token_secret)
```

Authorization in future R sessions

- The `create_token()` function should automatically save your token as an environment variable for you. So next time you start an R session [on the same machine], `rtweet` should automatically find your token.
- To make sure it works, restart your R session, run the following code, and again check to make sure the app name and `api_key` match.

```
## check to see if the token is loaded
library(rtweet)
get_token()
```

That's it!