Creating an OAuth Login Request

Section 2, Lecture 9

RESTful Web API Design with Node.js

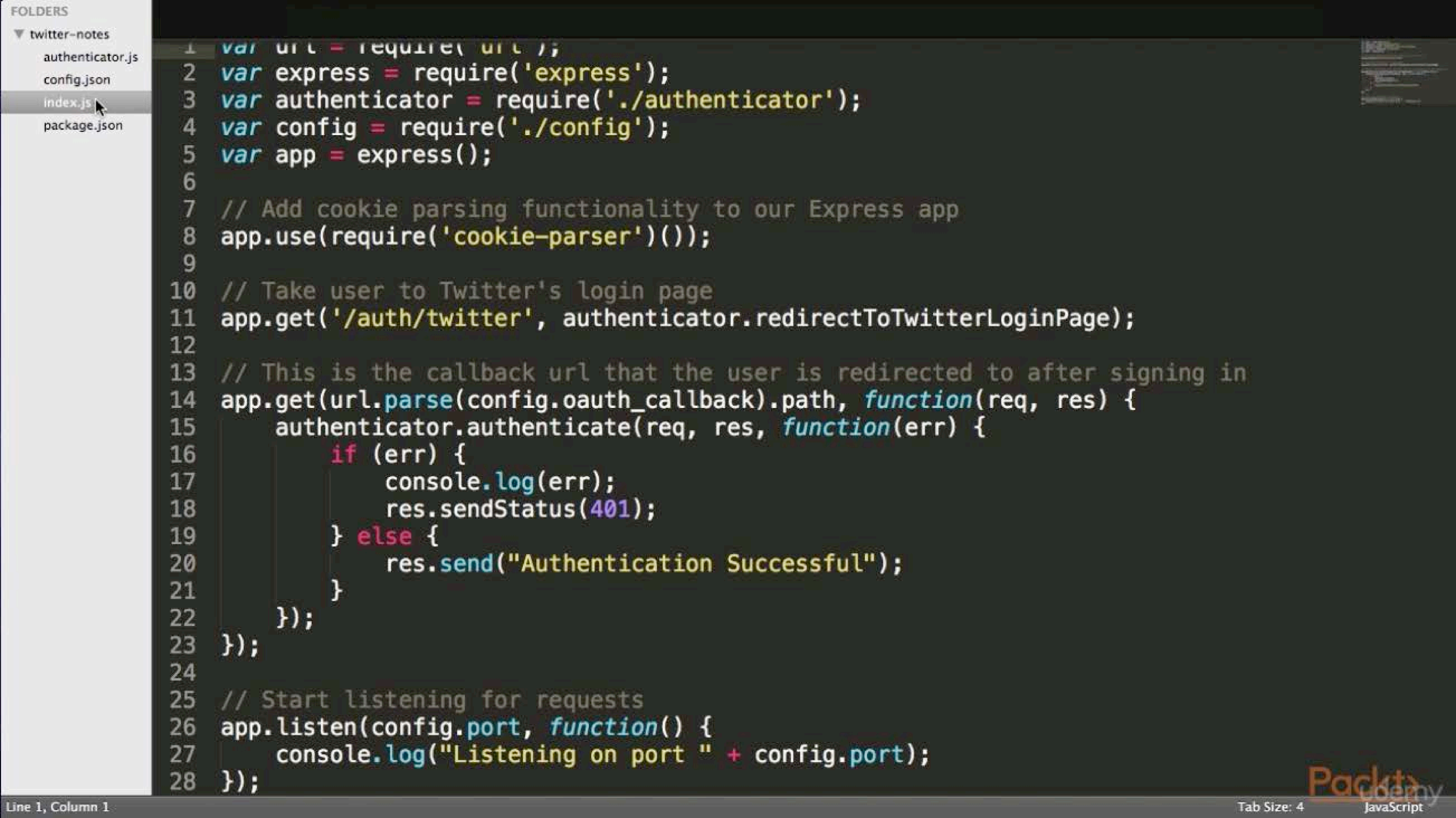
Saleh Hamadeh

Video 2.3

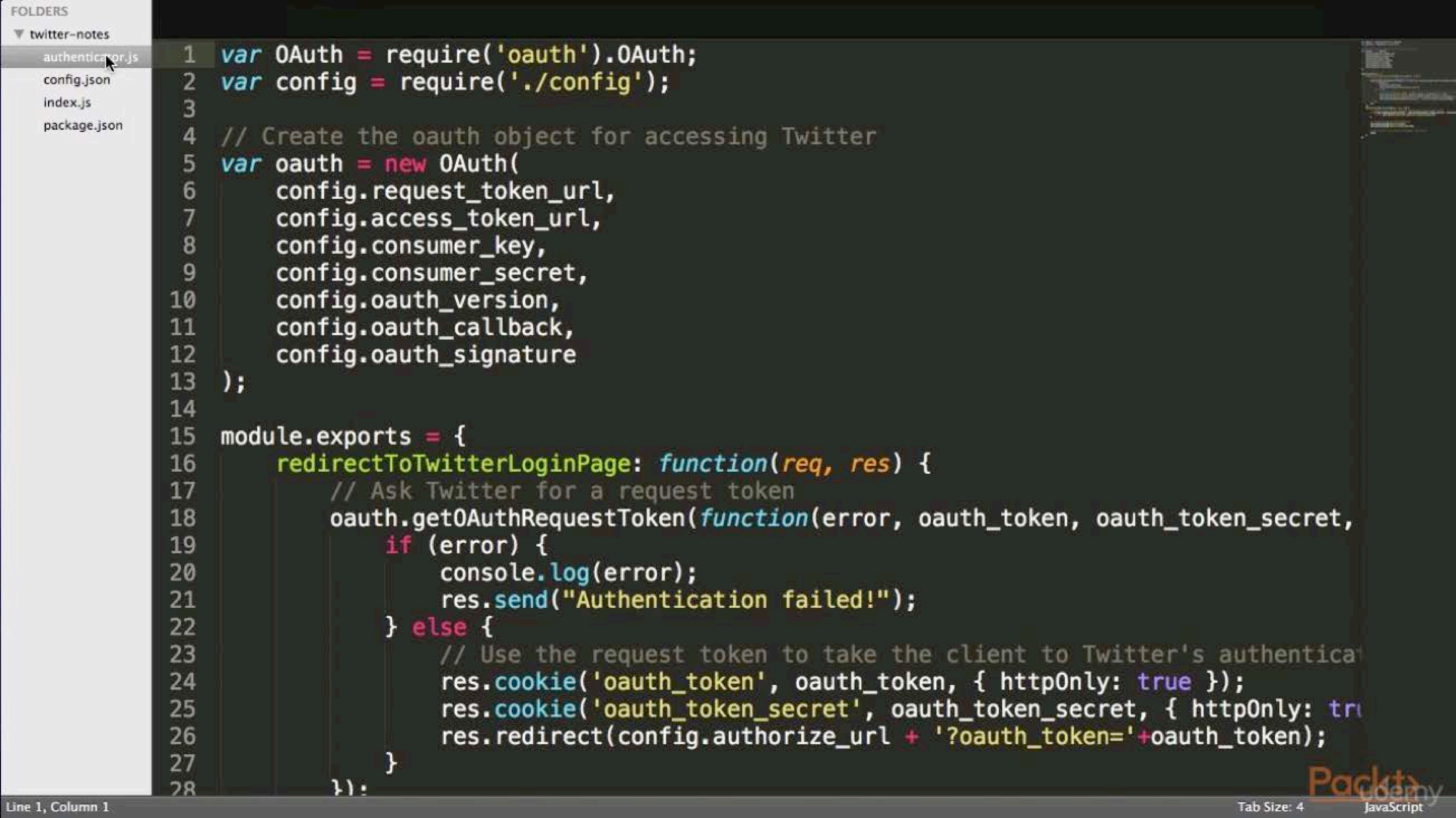
Creating an OAuth Login Request

In this Video, we are going to take a look at…

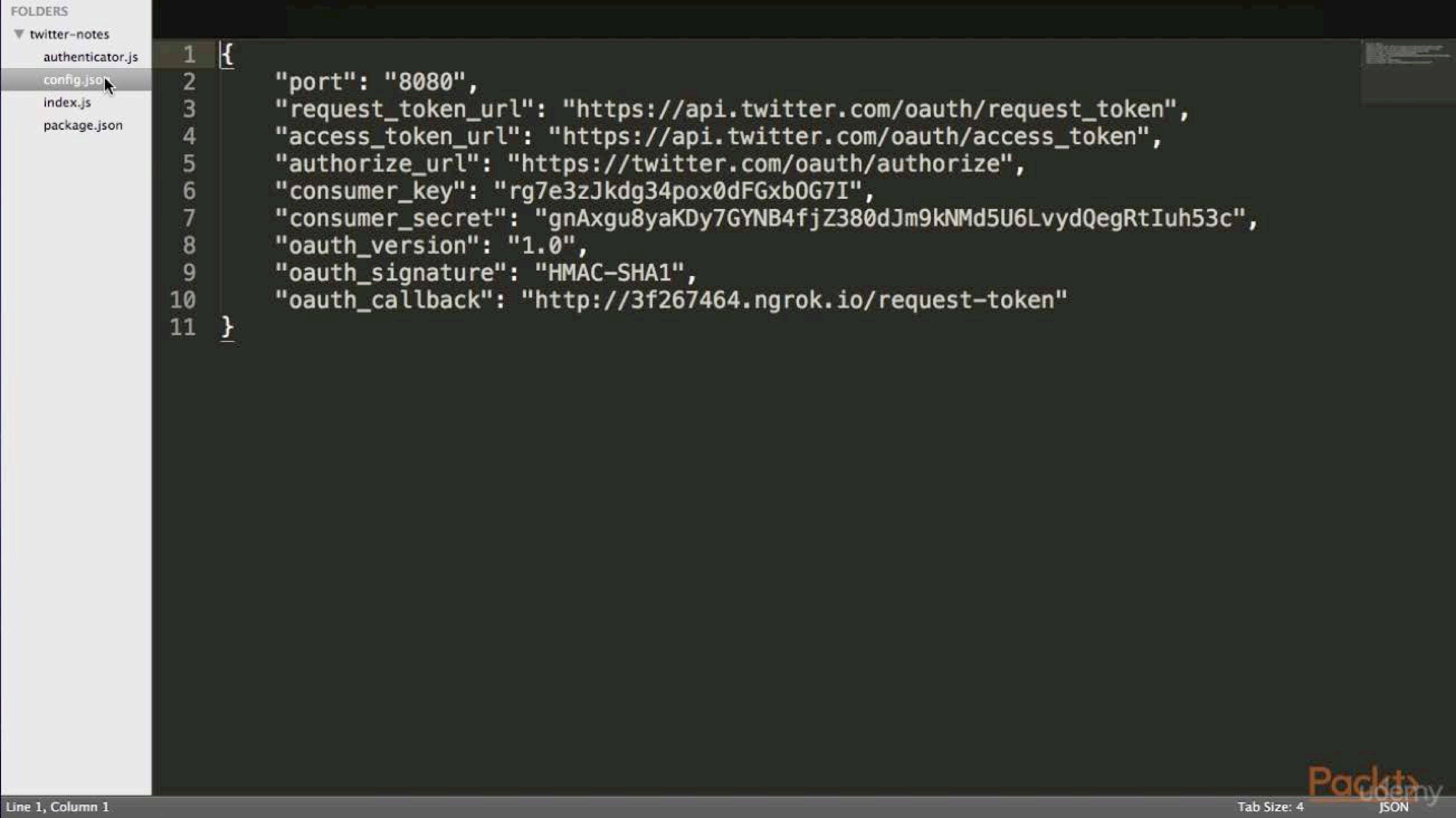
- Setting up the Express.js router
- Adding the Twitter's OAuth configuration
- Storing the state on the client
- Taking the user to the OAuth server



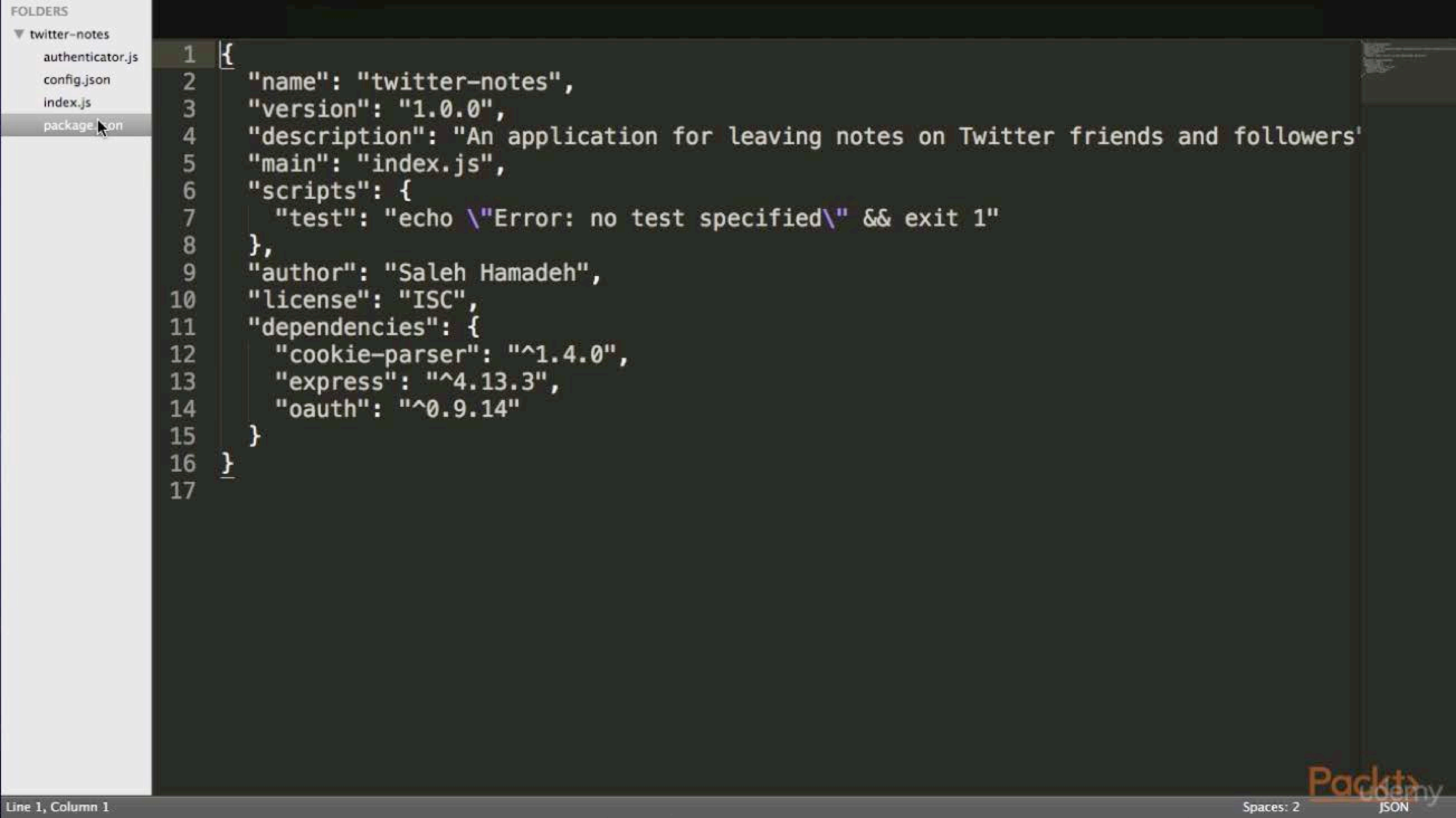
```javascript
var url = require('url');
var express = require('express');
var authenticator = require('./authenticator');
var config = require('./config');
var app = express();

// Add cookie parsing functionality to our Express app
app.use(require('cookie-parser')());

// Take user to Twitter's login page
app.get('/auth/twitter', authenticator.redirectToTwitterLoginPage);

// This is the callback url that the user is redirected to after signing in
app.get(url.parse(config.oauth_callback).path, function(req, res) {
    authenticator.authenticate(req, res, function(err) {
        if (err) {
            console.log(err);
            res.sendStatus(401);
        } else {
            res.send("Authentication Successful");
        }
    });
});

// Start listening for requests
app.listen(config.port, function() {
    console.log("Listening on port " + config.port);
});
```



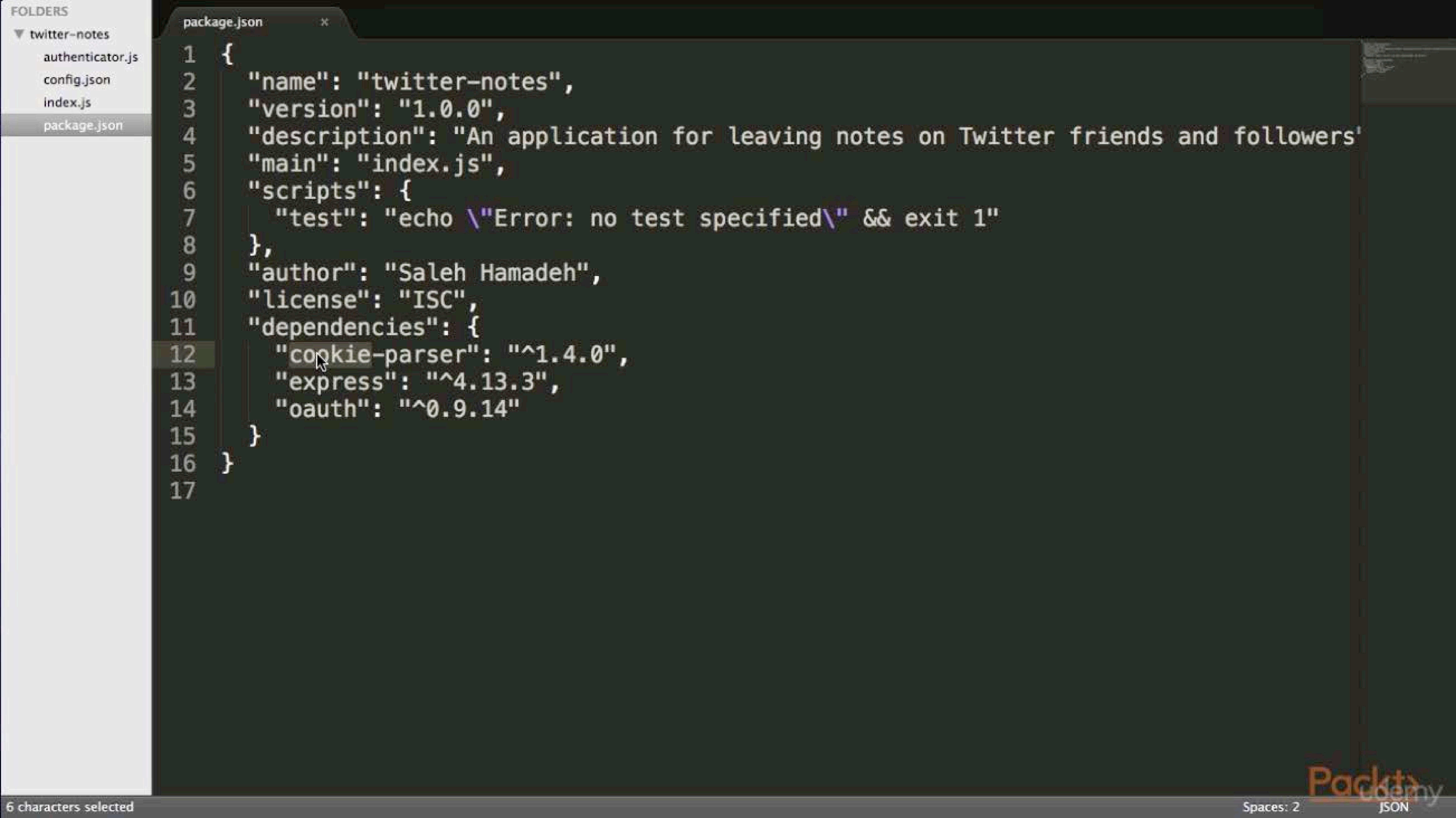
```javascript
var OAuth = require('oauth').OAuth;
var config = require('./config');

// Create the oauth object for accessing Twitter
var oauth = new OAuth(
    config.request_token_url,
    config.access_token_url,
    config.consumer_key,
    config.consumer_secret,
    config.oauth_version,
    config.oauth_callback,
    config.oauth_signature
);

module.exports = {
    redirectToTwitterLoginPage: function(req, res) {
        // Ask Twitter for a request token
        oauth.getOAuthRequestToken(function(error, oauth_token, oauth_token_secret,
            if (error) {
                console.log(error);
                res.send("Authentication failed!");
            } else {
                // Use the request token to take the client to Twitter's authentica
                res.cookie('oauth_token', oauth_token, { httpOnly: true });
                res.cookie('oauth_token_secret', oauth_token_secret, { httpOnly: tru
                res.redirect(config.authorize_url + '?oauth_token='+oauth_token);
            }
        });
```



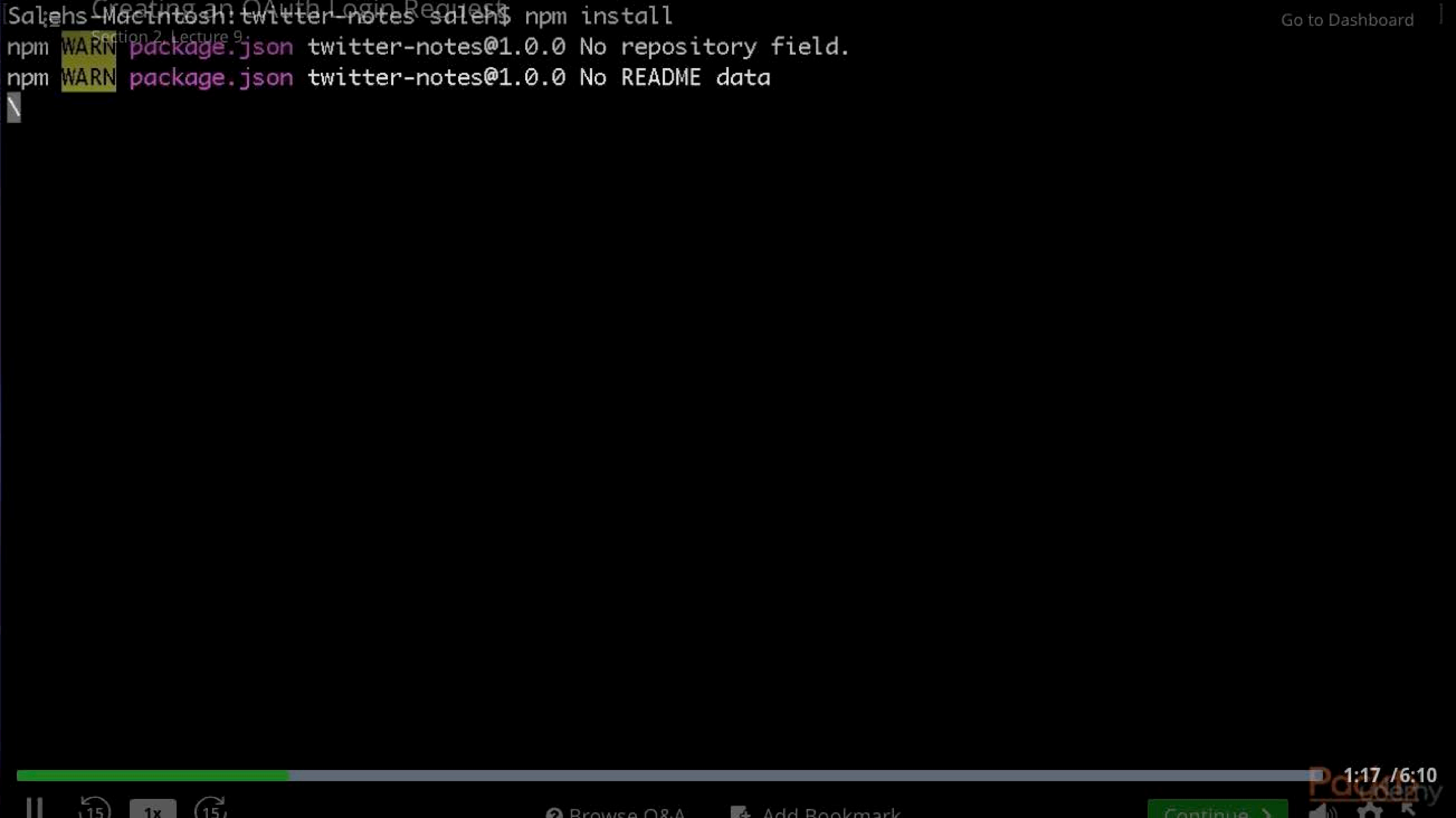
```json
{
    "port": "8080",
    "request_token_url": "https://api.twitter.com/oauth/request_token",
    "access_token_url": "https://api.twitter.com/oauth/access_token",
    "authorize_url": "https://twitter.com/oauth/authorize",
    "consumer_key": "rg7e3zJkdg34pox0dFGxbOG7I",
    "consumer_secret": "gnAxgu8yaKDy7GYNB4fjZ380dJm9kNMd5U6LvydQegRtIuh53c",
    "oauth_version": "1.0",
    "oauth_signature": "HMAC-SHA1",
    "oauth_callback": "http://3f267464.ngrok.io/request-token"
}
```



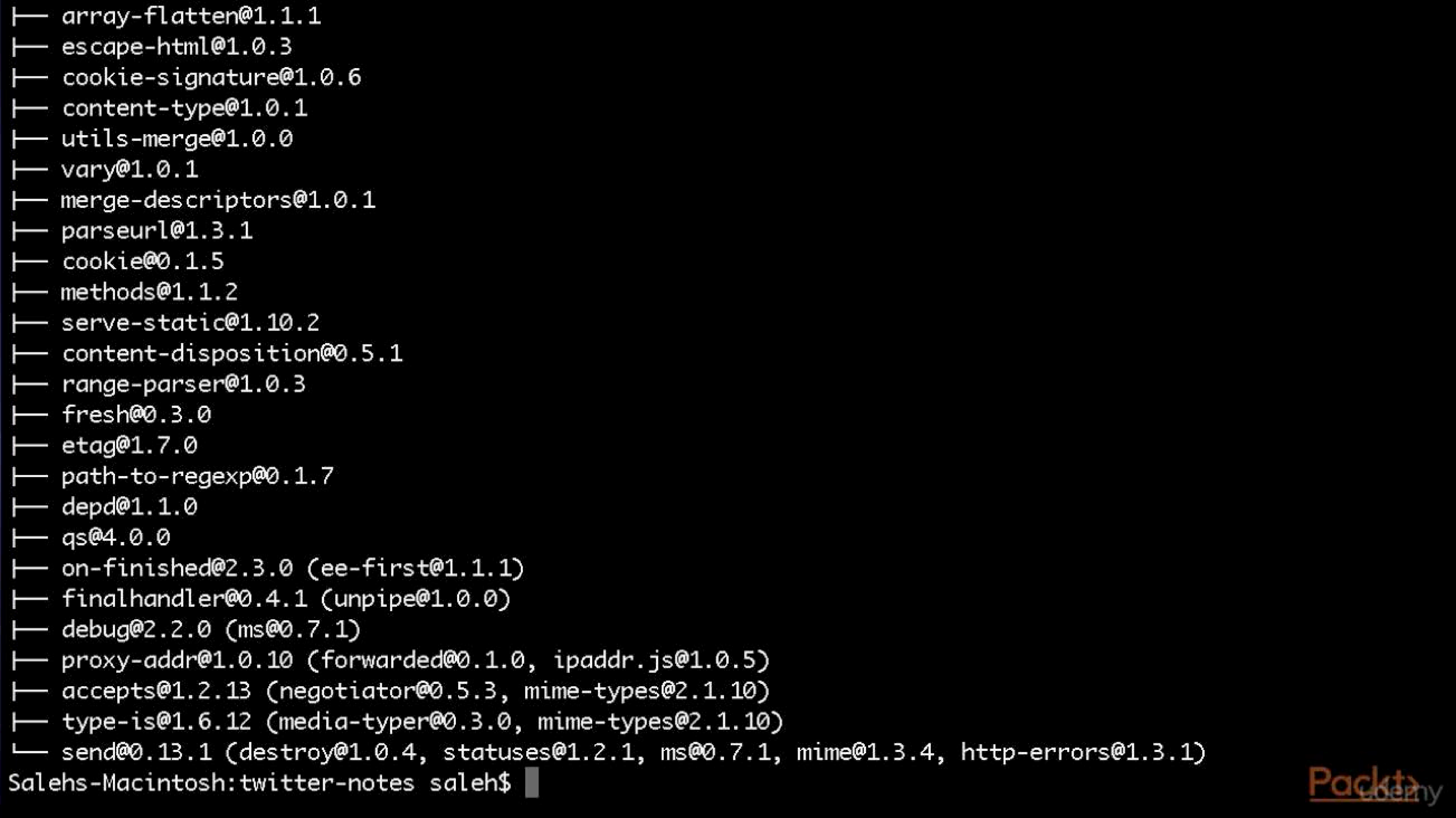
```json
{
  "name": "twitter-notes",
  "version": "1.0.0",
  "description": "An application for leaving notes on Twitter friends and followers"
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Saleh Hamadeh",
  "license": "ISC",
  "dependencies": {
    "cookie-parser": "^1.4.0",
    "express": "^4.13.3",
    "oauth": "^0.9.14"
  }
}
```



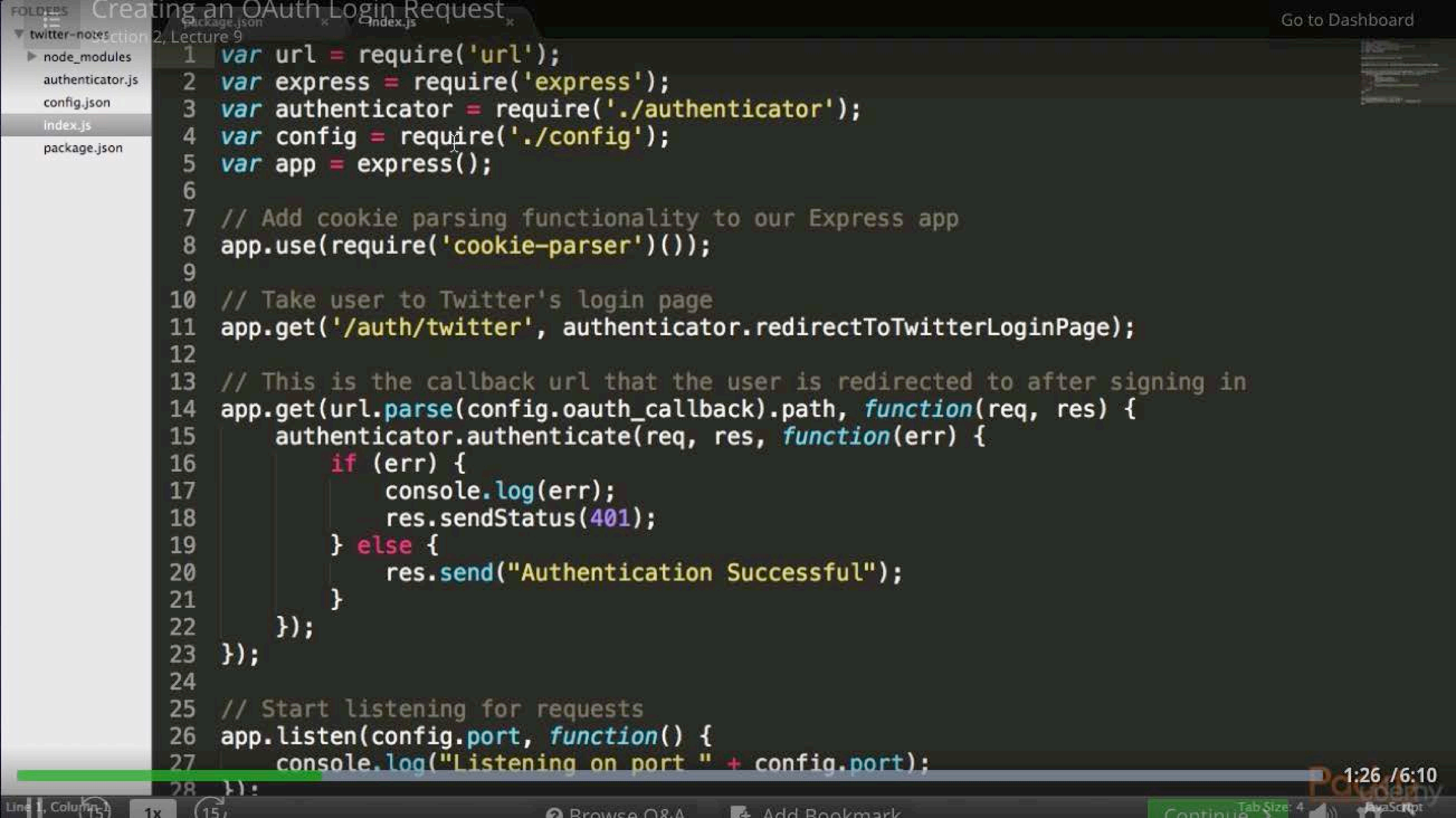
```json
{
  "name": "twitter-notes",
  "version": "1.0.0",
  "description": "An application for leaving notes on Twitter friends and followers"
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Saleh Hamadeh",
  "license": "ISC",
  "dependencies": {
    "cookie-parser": "^1.4.0",
    "express": "^4.13.3",
    "oauth": "^0.9.14"
  }
}

```



```
Salehs-Macintosh:twitter-notes saleh$ npm install
npm WARN package.json twitter-notes@1.0.0 No repository field.
npm WARN package.json twitter-notes@1.0.0 No README data
\
```



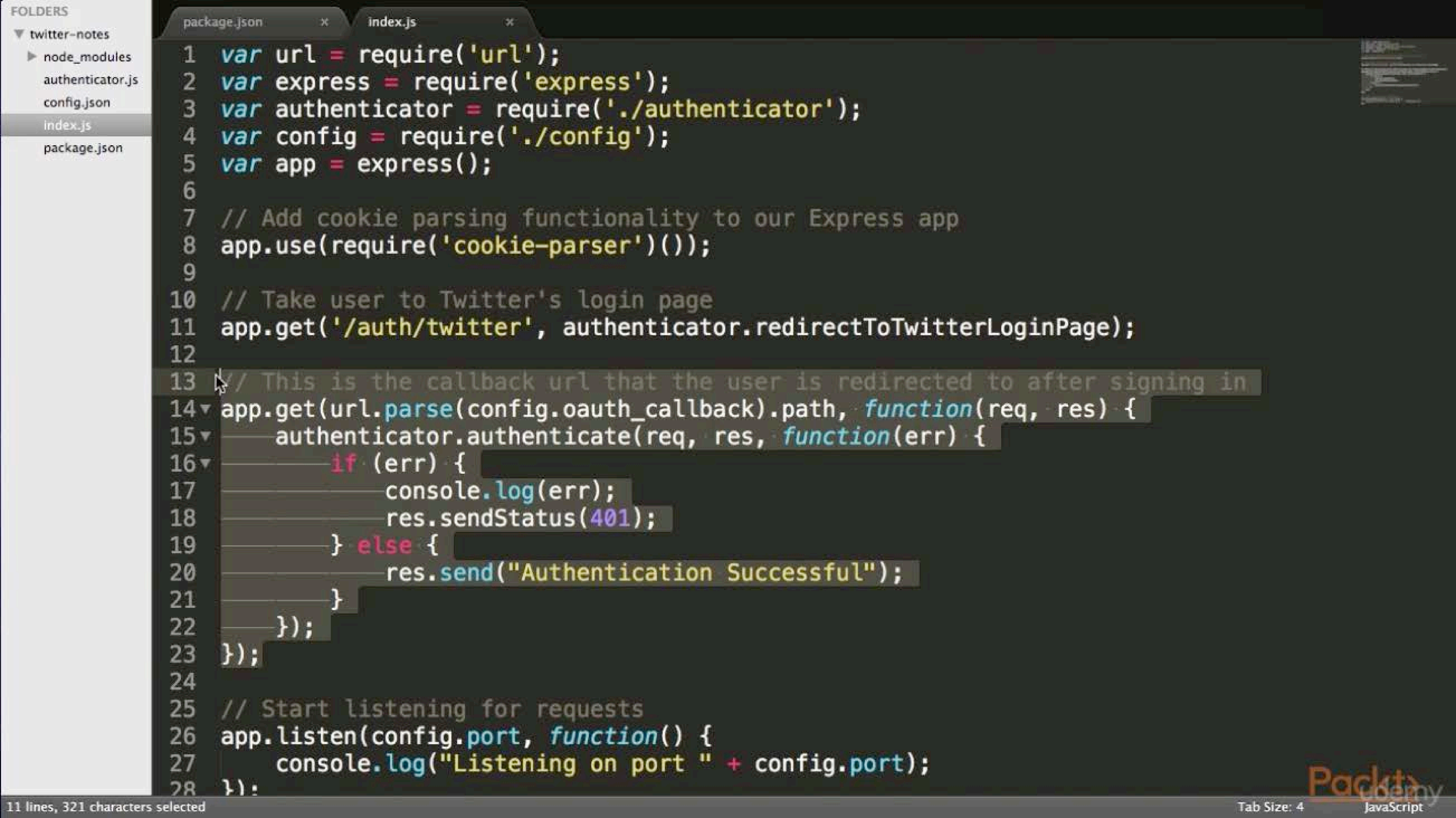
```
├── array-flatten@1.1.1
├── escape-html@1.0.3
├── cookie-signature@1.0.6
├── content-type@1.0.1
├── utils-merge@1.0.0
├── vary@1.0.1
├── merge-descriptors@1.0.1
├── parseurl@1.3.1
├── cookie@0.1.5
├── methods@1.1.2
├── serve-static@1.10.2
├── content-disposition@0.5.1
├── range-parser@1.0.3
├── fresh@0.3.0
├── etag@1.7.0
├── path-to-regexp@0.1.7
├── depd@1.1.0
├── qs@4.0.0
├── on-finished@2.3.0 (ee-first@1.1.1)
├── finalhandler@0.4.1 (unpipe@1.0.0)
├── debug@2.2.0 (ms@0.7.1)
├── proxy-addr@1.0.10 (forwarded@0.1.0, ipaddr.js@1.0.5)
├── accepts@1.2.13 (negotiator@0.5.3, mime-types@2.1.10)
├── type-is@1.6.12 (media-typer@0.3.0, mime-types@2.1.10)
└── send@0.13.1 (destroy@1.0.4, statuses@1.2.1, ms@0.7.1, mime@1.3.4, http-errors@1.3.1)
Salehs-Macintosh:twitter-notes saleh$
```



Creating an OAuth Login Request

Section 2, Lecture 9

```javascript
var url = require('url');
var express = require('express');
var authenticator = require('./authenticator');
var config = require('./config');
var app = express();

// Add cookie parsing functionality to our Express app
app.use(require('cookie-parser')());

// Take user to Twitter's login page
app.get('/auth/twitter', authenticator.redirectToTwitterLoginPage);

// This is the callback url that the user is redirected to after signing in
app.get(url.parse(config.oauth_callback).path, function(req, res) {
    authenticator.authenticate(req, res, function(err) {
        if (err) {
            console.log(err);
            res.sendStatus(401);
        } else {
            res.send("Authentication Successful");
        }
    });
});

// Start listening for requests
app.listen(config.port, function() {
    console.log("Listening on port " + config.port);
});
```

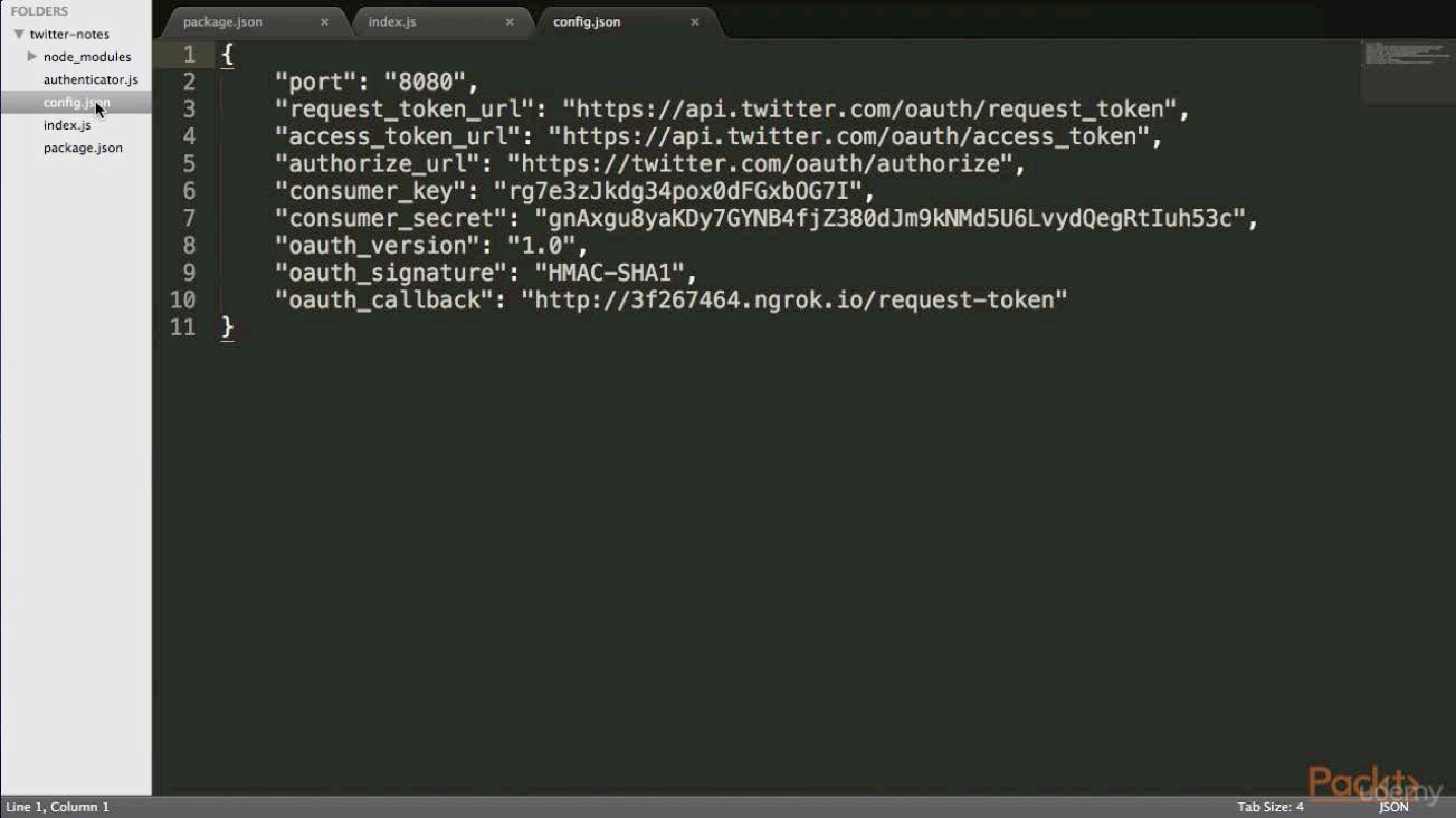


```javascript
var url = require('url');
var express = require('express');
var authenticator = require('./authenticator');
var config = require('./config');
var app = express();

// Add cookie parsing functionality to our Express app
app.use(require('cookie-parser')());

// Take user to Twitter's login page
app.get('/auth/twitter', authenticator.redirectToTwitterLoginPage);

// This is the callback url that the user is redirected to after signing in
app.get(url.parse(config.oauth_callback).path, function(req, res) {
    authenticator.authenticate(req, res, function(err) {
        if (err) {
            console.log(err);
            res.sendStatus(401);
        } else {
            res.send("Authentication Successful");
        }
    });
});

// Start listening for requests
app.listen(config.port, function() {
    console.log("Listening on port " + config.port);
});
```

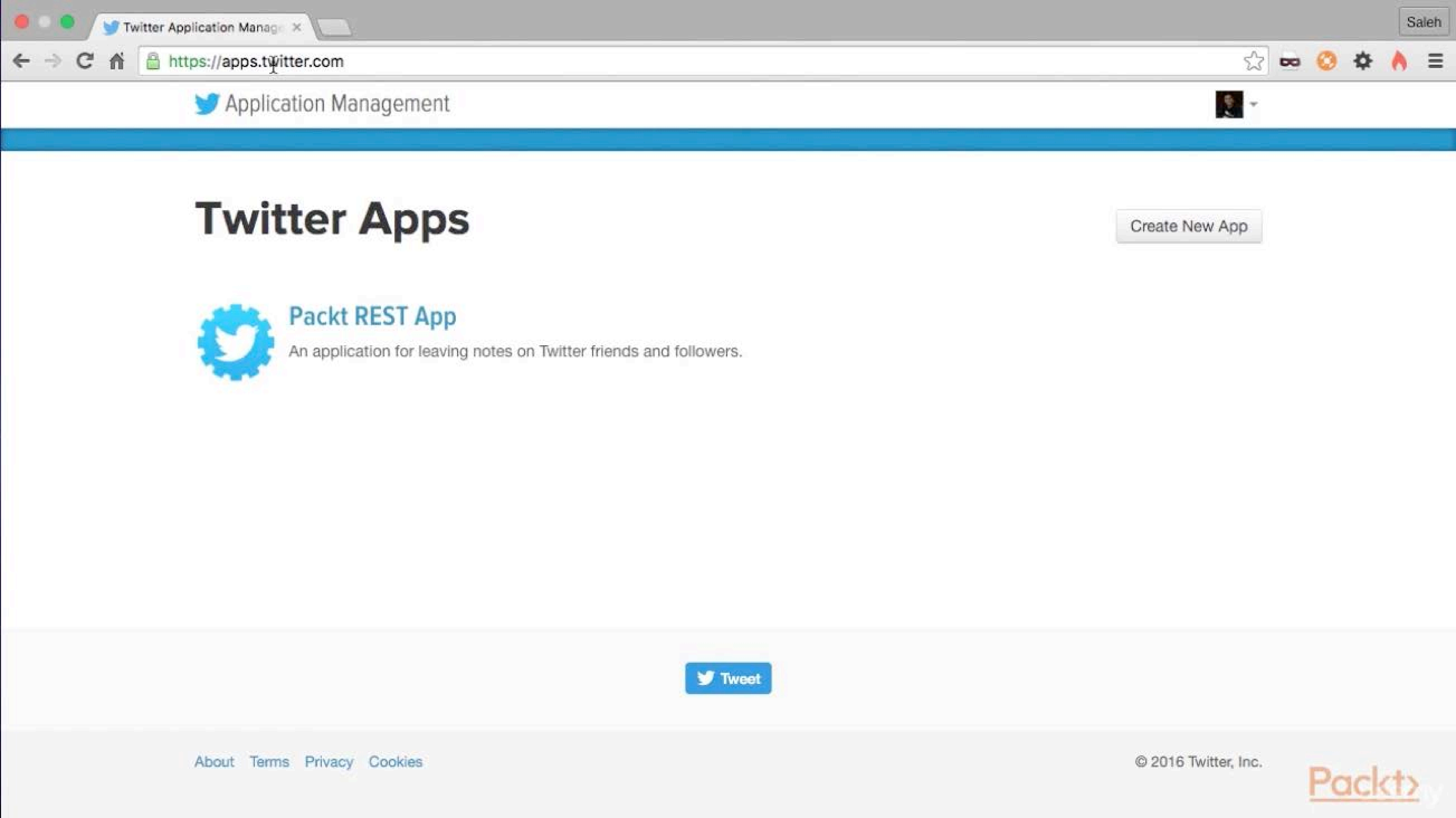
11 lines, 321 characters selected

Tab Size: 4

JavaScript



```
{
    "port": "8080",
    "request_token_url": "https://api.twitter.com/oauth/request_token",
    "access_token_url": "https://api.twitter.com/oauth/access_token",
    "authorize_url": "https://twitter.com/oauth/authorize",
    "consumer_key": "rg7e3zJkdg34pox0dFGxbOG7I",
    "consumer_secret": "gnAxgu8yaKDy7GYNB4fjZ380dJm9kNMd5U6LvydQegRtIuh53c",
    "oauth_version": "1.0",
    "oauth_signature": "HMAC-SHA1",
    "oauth_callback": "http://3f267464.ngrok.io/request-token"
}
```

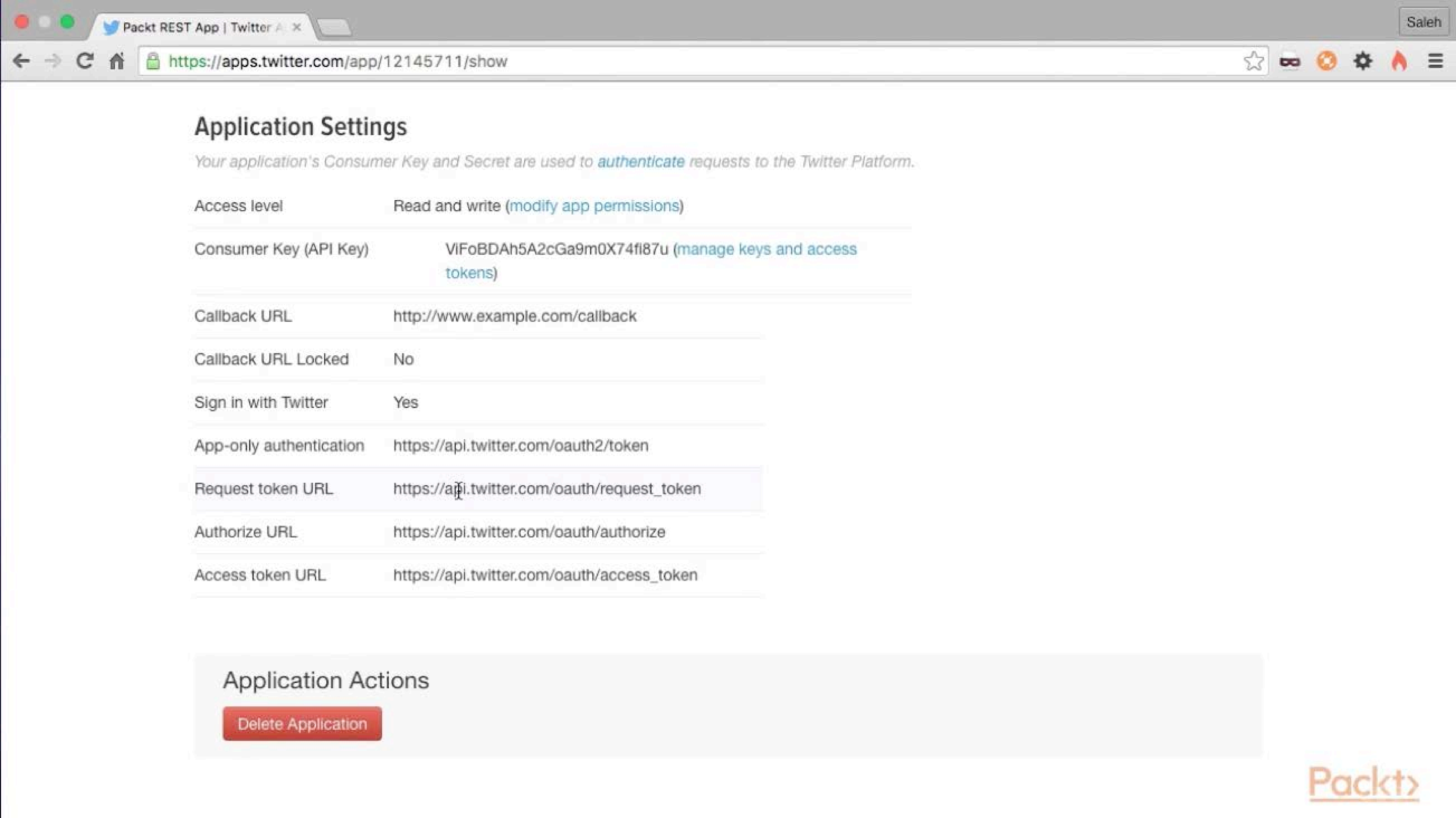


Twitter Apps

Packt REST App

An application for leaving notes on Twitter friends and followers.

© 2016 Twitter, Inc.

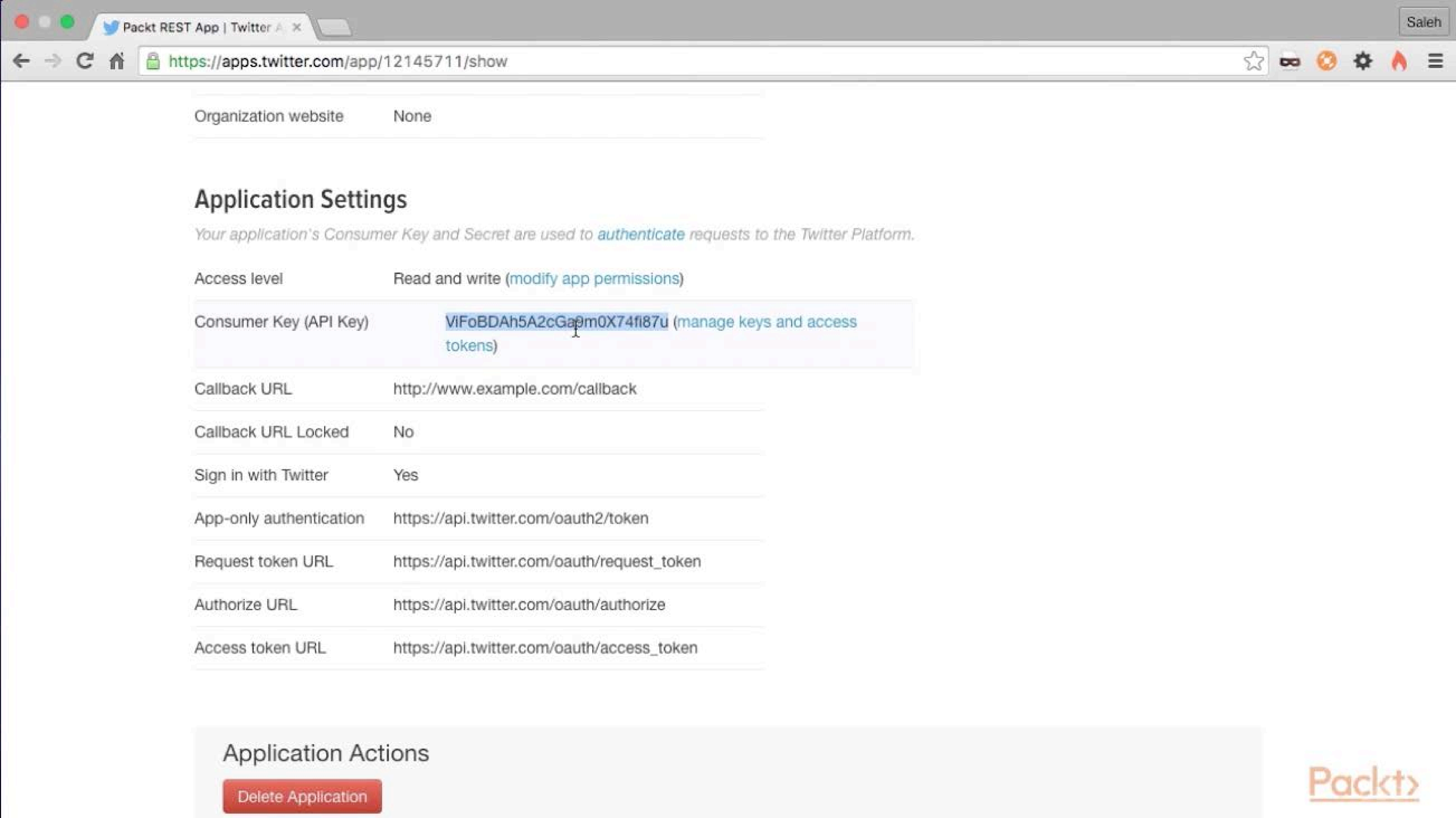

Packt REST App | Twitter A
https://apps.twitter.com/app/12145711/show
Saleh

Application Settings

Your application's Consumer Key and Secret are used to authenticate requests to the Twitter Platform.

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	ViFoBDAh5A2cGa9m0X74fi87u (manage keys and access tokens)
Callback URL	http://www.example.com/callback
Callback URL Locked	No
Sign in with Twitter	Yes
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token

Application Actions



Packt REST App | Twitter A
https://apps.twitter.com/app/12145711/show
Saleh

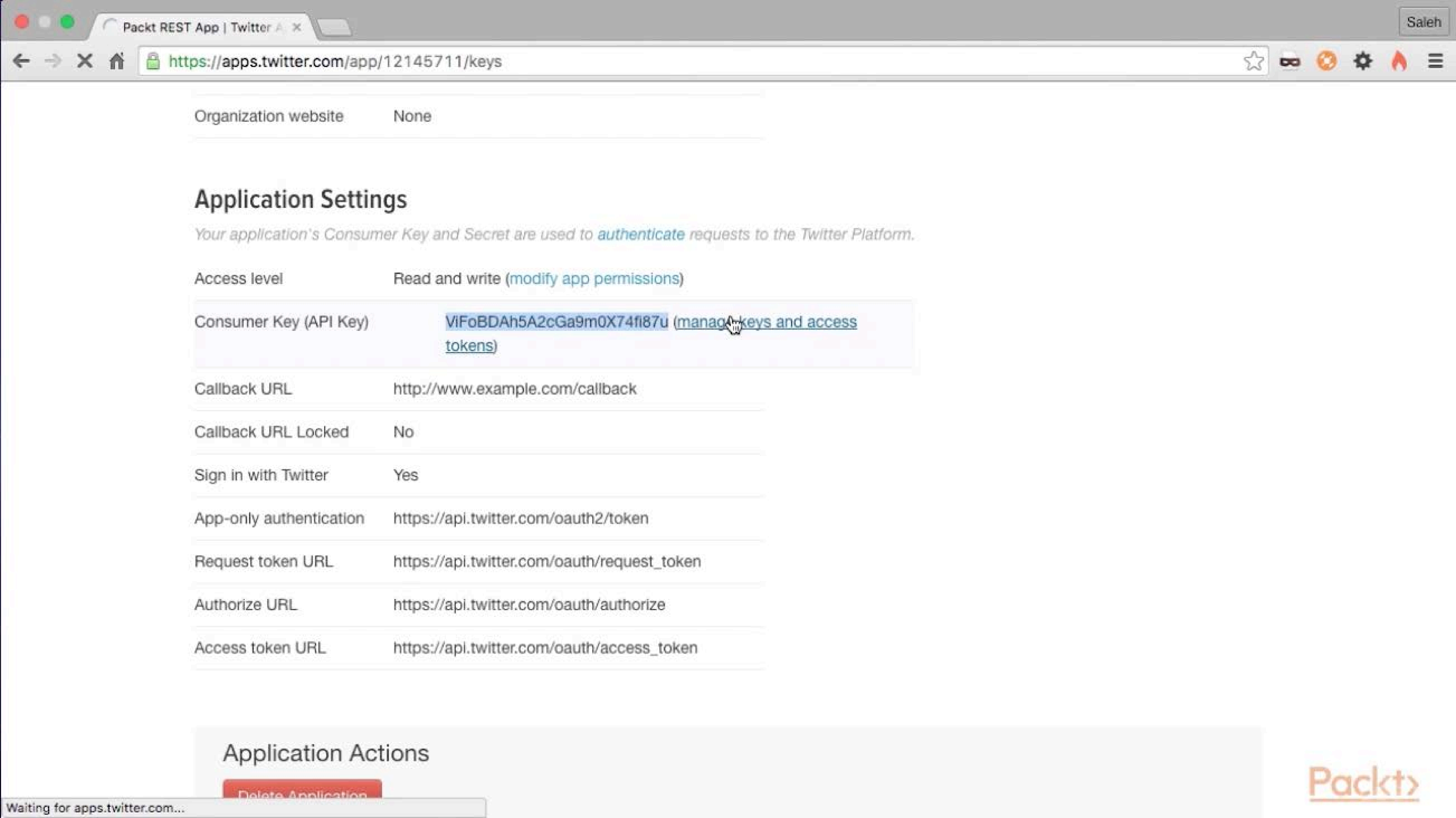
Organization website	None

Application Settings

Your application's Consumer Key and Secret are used to authenticate requests to the Twitter Platform.

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	ViFoBDAh5A2cGa9m0X74fi87u (manage keys and access tokens)
Callback URL	http://www.example.com/callback
Callback URL Locked	No
Sign in with Twitter	Yes
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token

Application Actions


Packt REST App | Twitter A
Saleh
https://apps.twitter.com/app/12145711/keys
Waiting for apps.twitter.com...
Delete Application

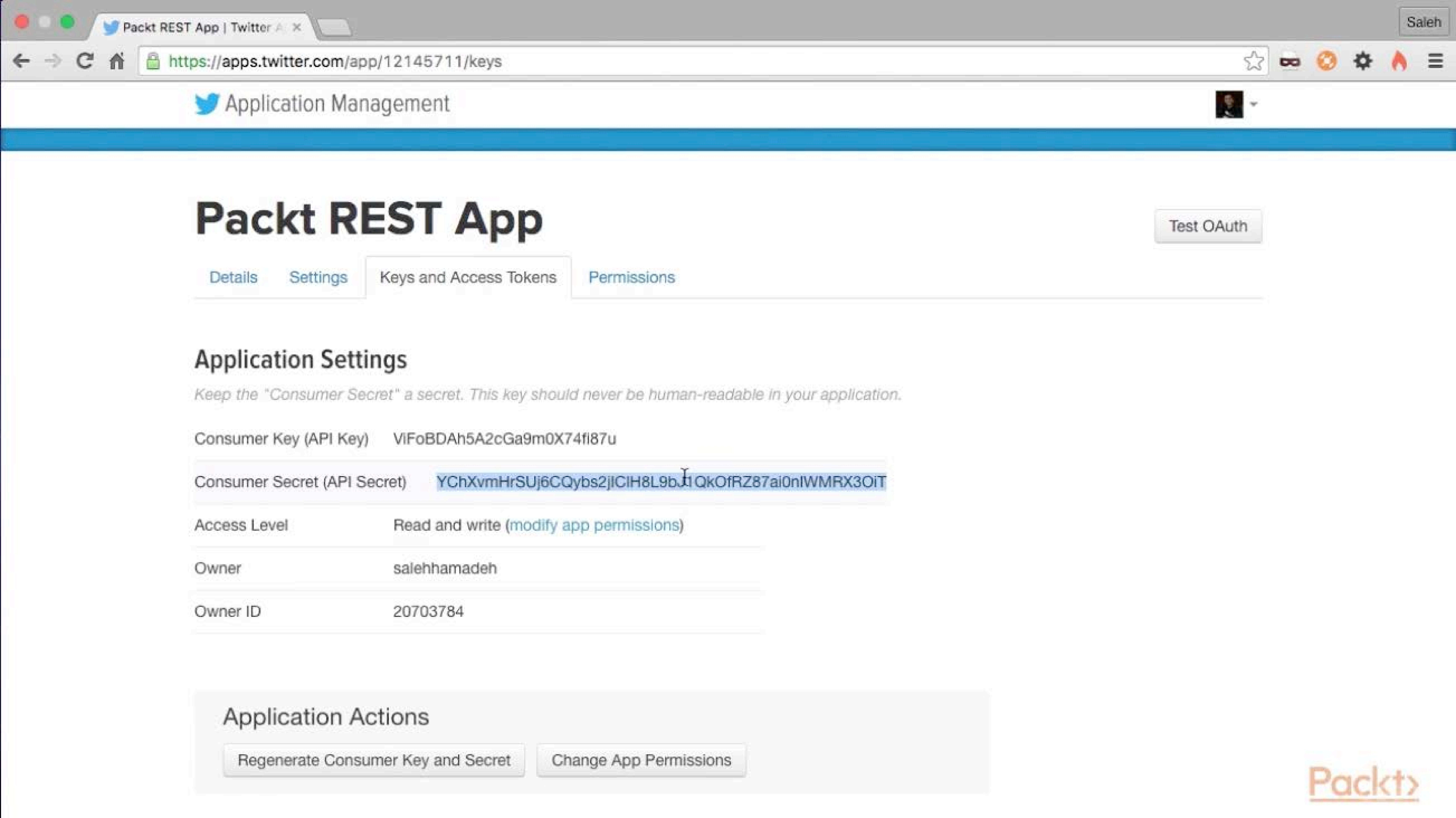
Organization website	None

Application Settings

Your application's Consumer Key and Secret are used to authenticate requests to the Twitter Platform.

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	ViFoBDAh5A2cGa9m0X74fi87u (manage keys and access tokens)
Callback URL	http://www.example.com/callback
Callback URL Locked	No
Sign in with Twitter	Yes
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token

Application Actions


Packt REST App | Twitter A
https://apps.twitter.com/app/12145711/keys
Saleh
Application Management

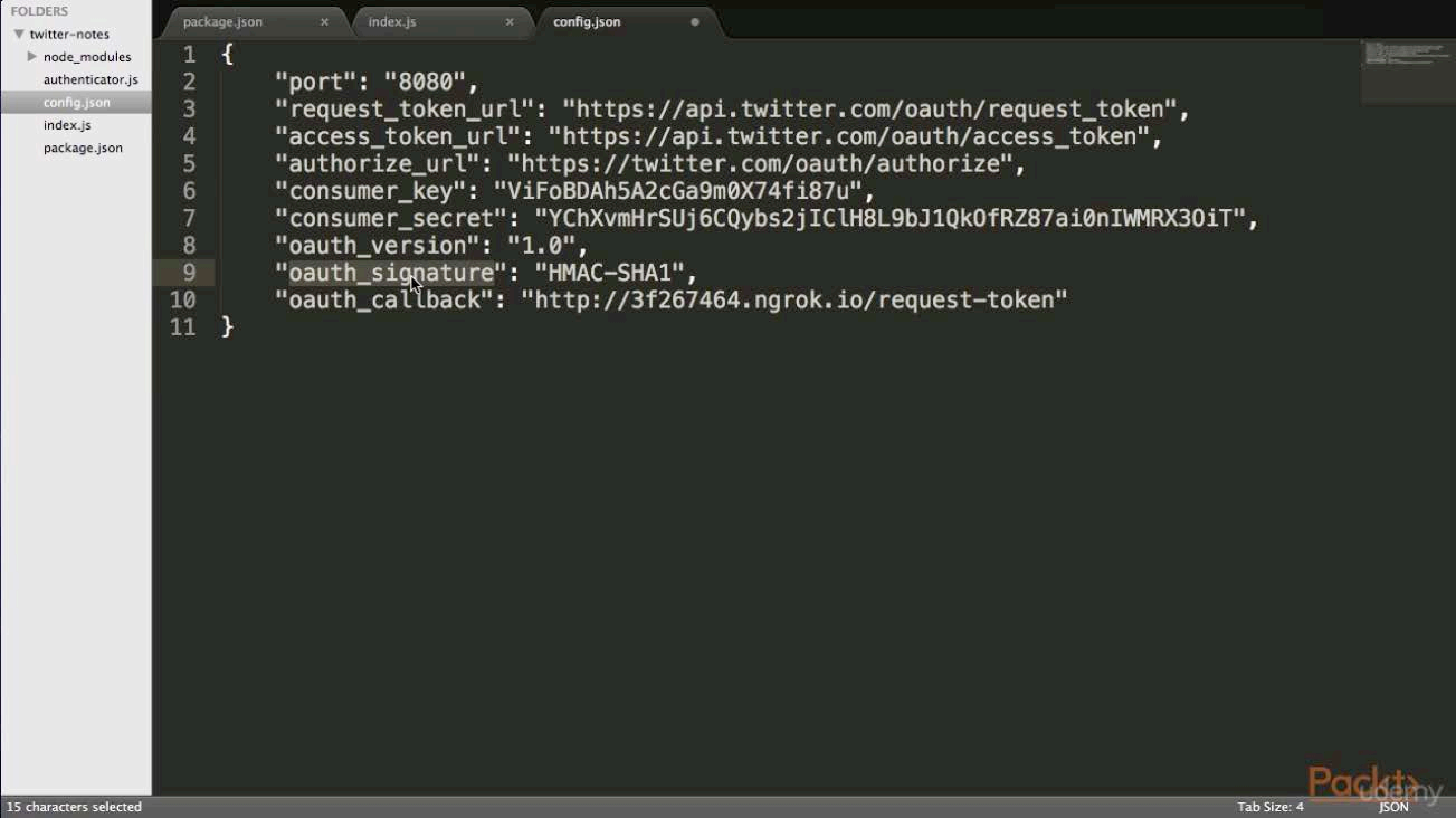
Packt REST App

Application Settings

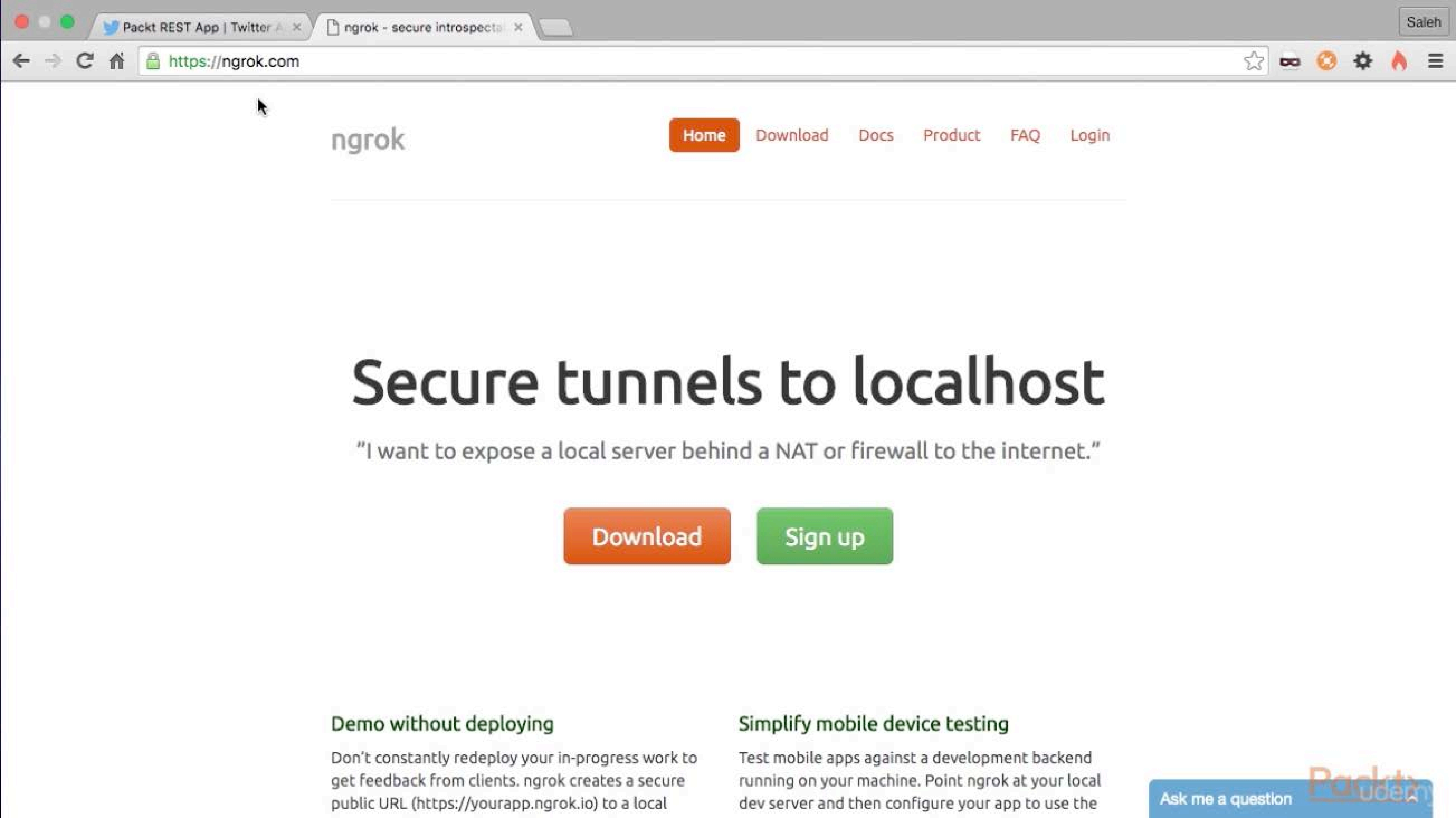
Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	ViFoBDAh5A2cGa9m0X74fl87u
Consumer Secret (API Secret)	YChXvmHrSUj6CQybs2jlClH8L9bJ1QkOfRZ87ai0nlWMRX3OiT
Access Level	Read and write (modify app permissions)
Owner	salehhamadeh
Owner ID	20703784

Application Actions



```json
{
    "port": "8080",
    "request_token_url": "https://api.twitter.com/oauth/request_token",
    "access_token_url": "https://api.twitter.com/oauth/access_token",
    "authorize_url": "https://twitter.com/oauth/authorize",
    "consumer_key": "ViFoBDAh5A2cGa9m0X74fi87u",
    "consumer_secret": "YChXvmHrSUj6CQybs2jIClH8L9bJ1QkOfRZ87ai0nIWMRX3OiT",
    "oauth_version": "1.0",
    "oauth_signature": "HMAC-SHA1",
    "oauth_callback": "http://3f267464.ngrok.io/request-token"
}
```



ngrok

Secure tunnels to localhost

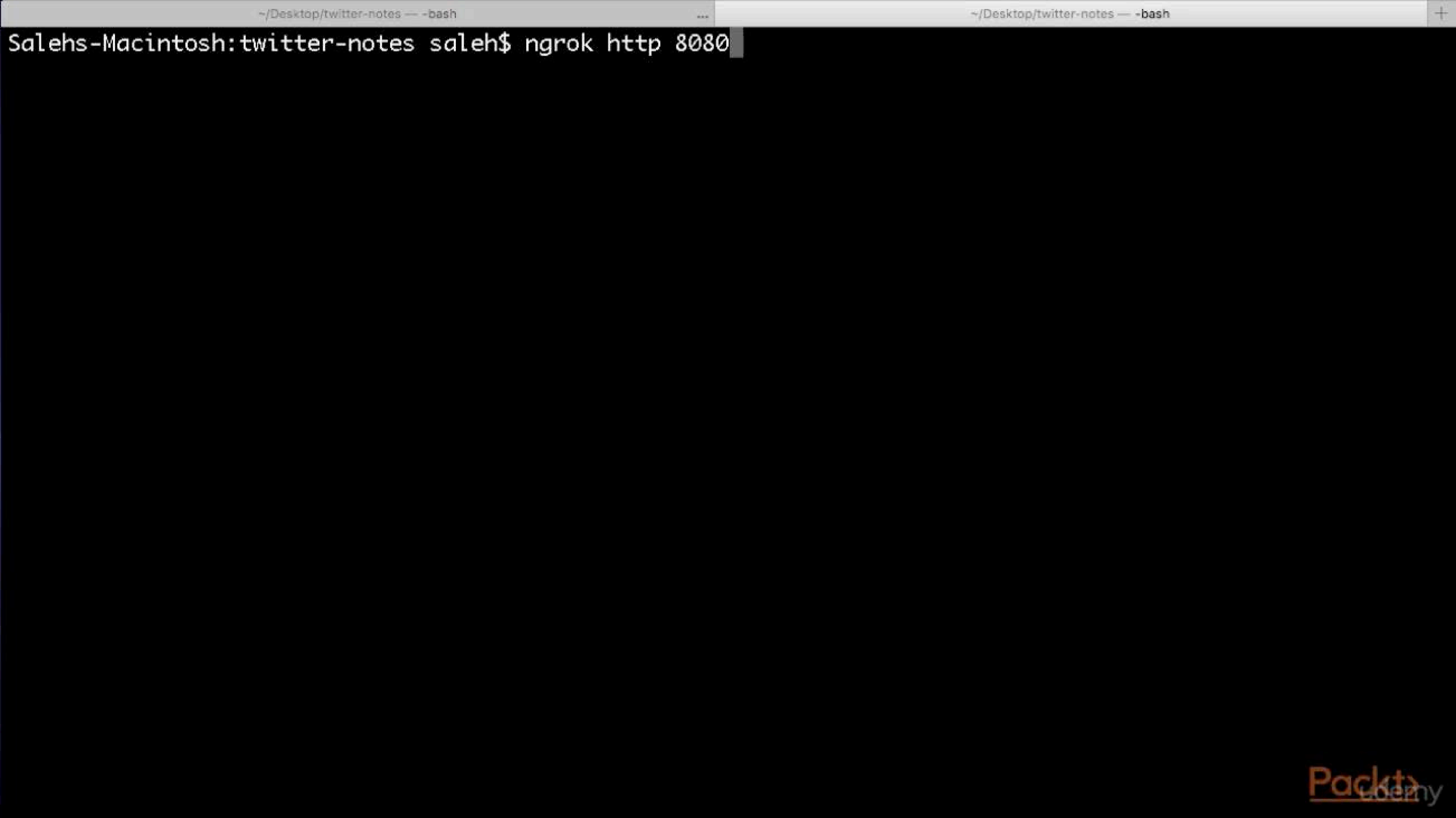
"I want to expose a local server behind a NAT or firewall to the internet."

Demo without deploying

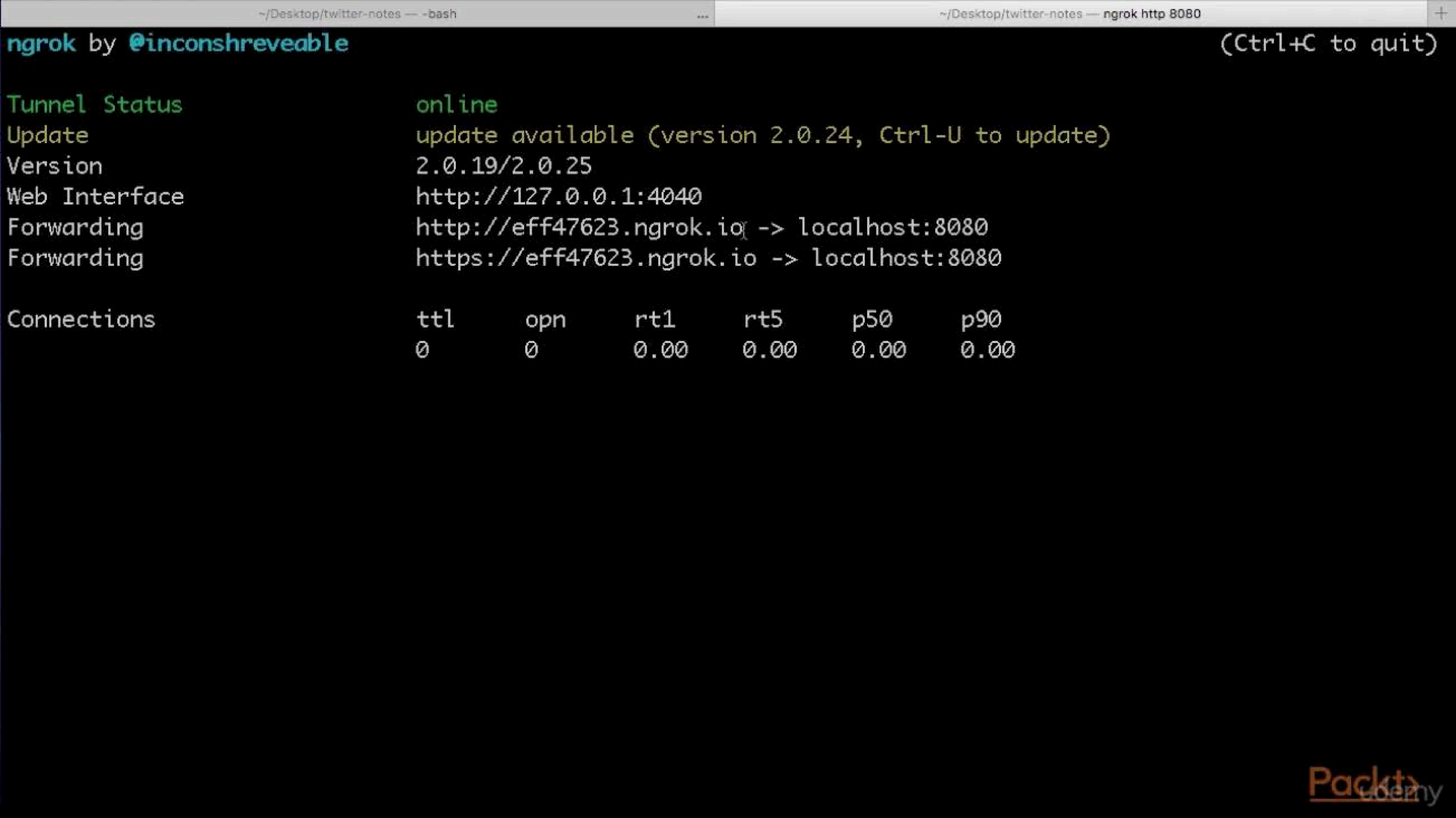
Don't constantly redeploy your in-progress work to get feedback from clients. ngrok creates a secure public URL (https://yourapp.ngrok.io) to a local

Simplify mobile device testing

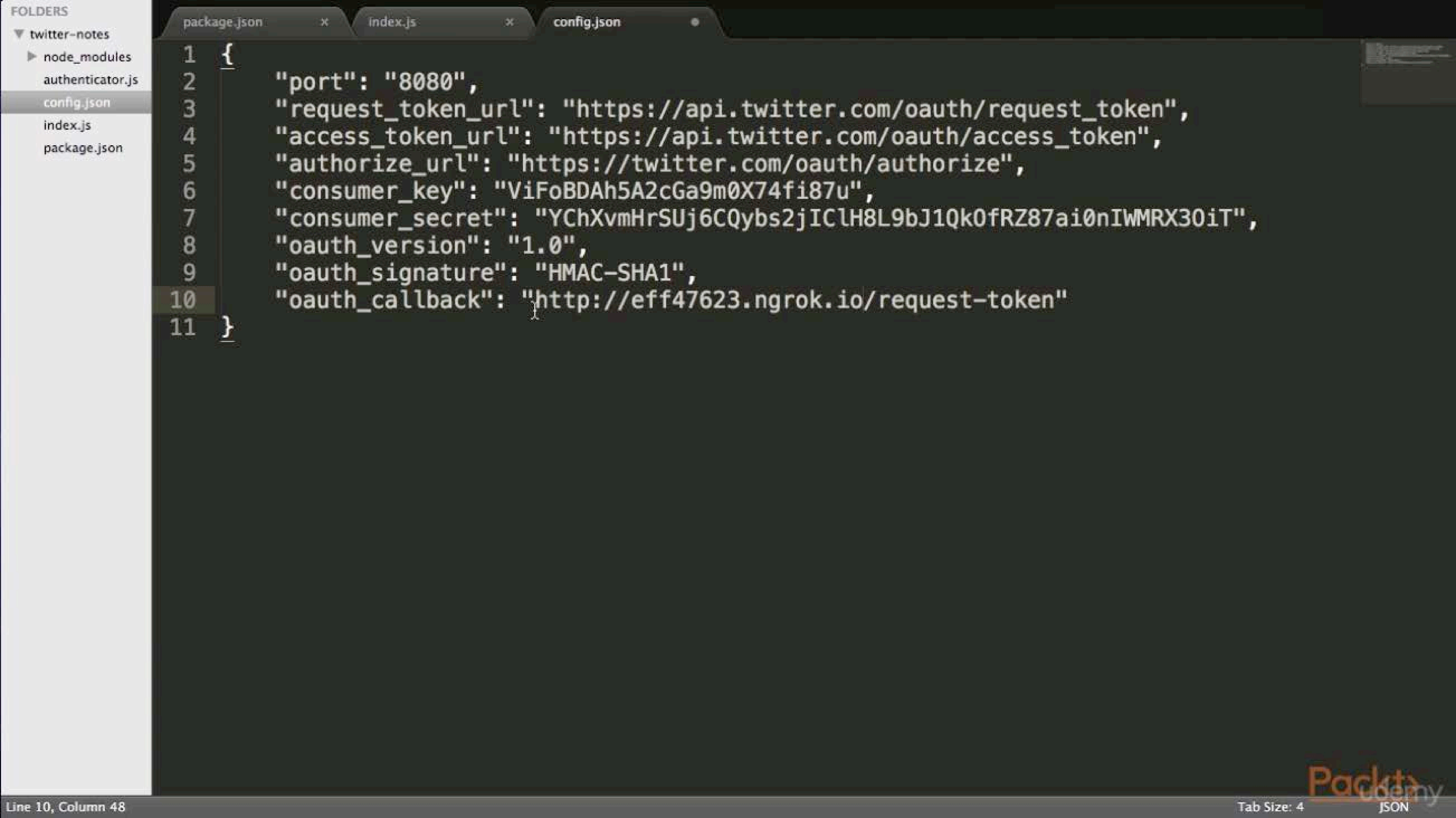
Test mobile apps against a development backend running on your machine. Point ngrok at your local dev server and then configure your app to use the


~/Desktop/twitter-notes — -bash
~/Desktop/twitter-notes — -bash

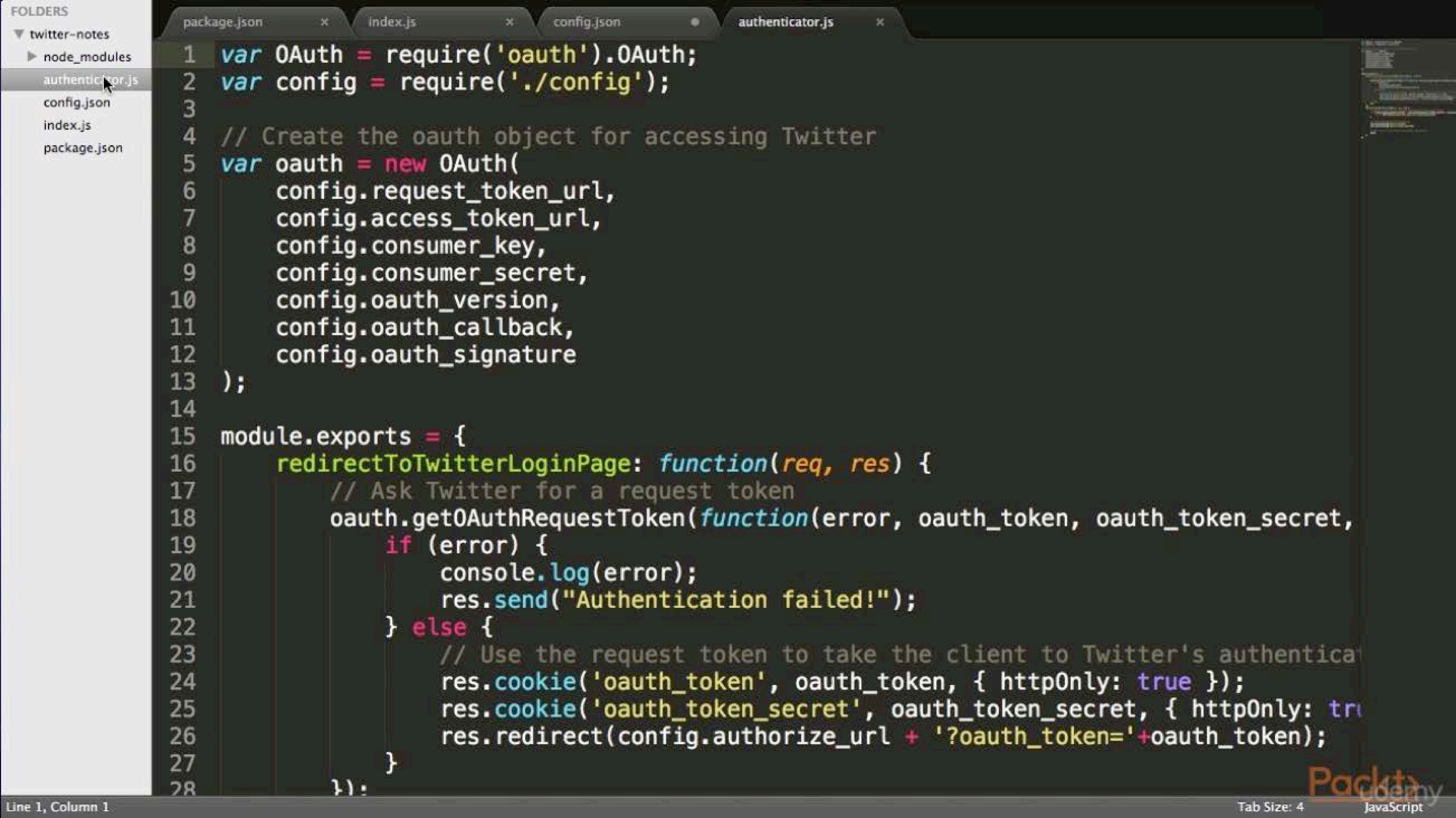
```
Salehs-Macintosh:twitter-notes saleh$ ngrok http 8080
```


~/Desktop/twitter-notes — -bash
~/Desktop/twitter-notes — ngrok http 8080

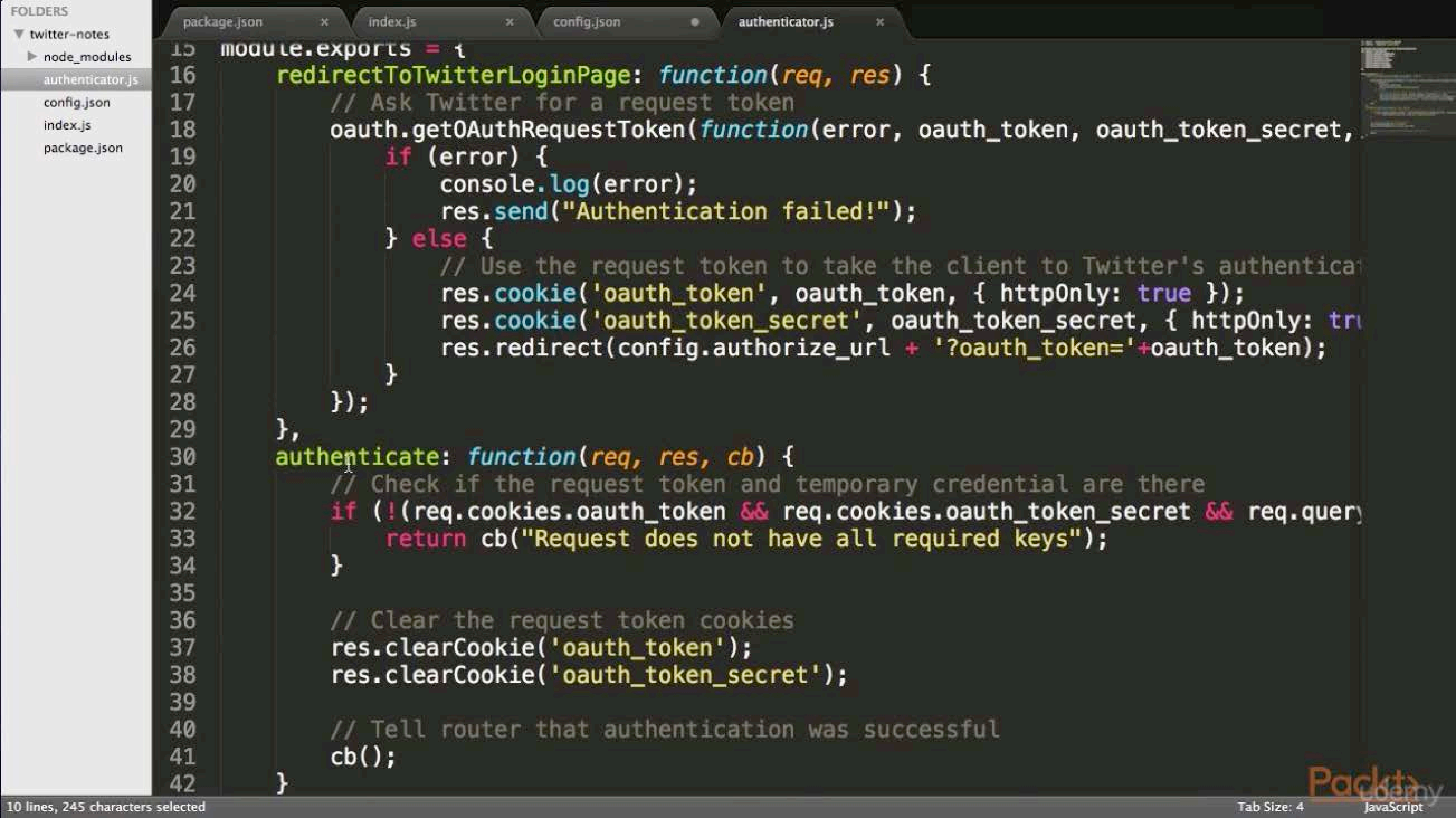
```
ngrok by @inconshreveable                                                                    (Ctrl+C to quit)

Tunnel Status                 online
Update                        update available (version 2.0.24, Ctrl-U to update)
Version                       2.0.19/2.0.25
Web Interface                 http://127.0.0.1:4040
Forwarding                    http://eff47623.ngrok.io -> localhost:8080
Forwarding                    https://eff47623.ngrok.io -> localhost:8080

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```



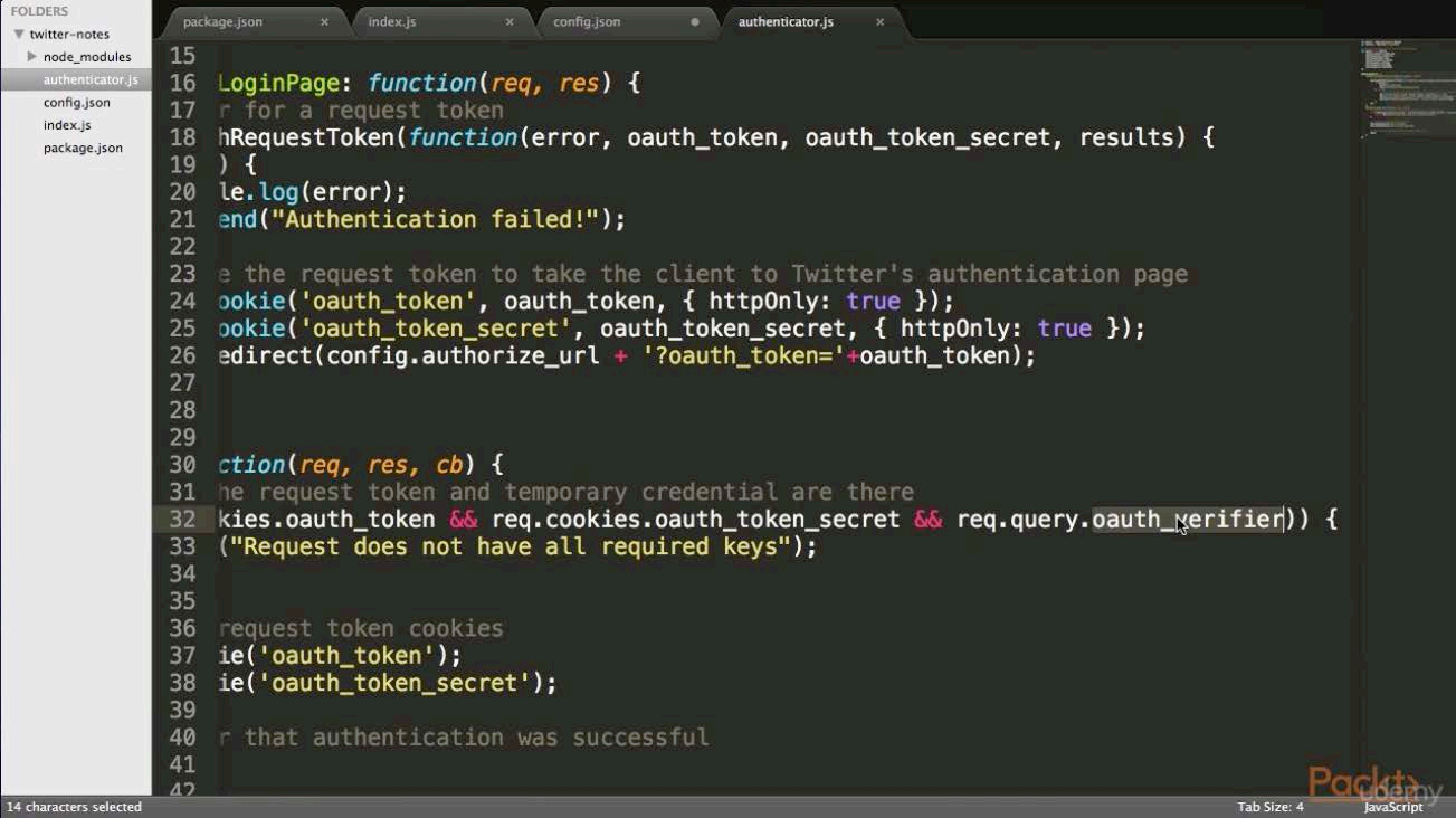
```
{
    "port": "8080",
    "request_token_url": "https://api.twitter.com/oauth/request_token",
    "access_token_url": "https://api.twitter.com/oauth/access_token",
    "authorize_url": "https://twitter.com/oauth/authorize",
    "consumer_key": "ViFoBDAh5A2cGa9m0X74fi87u",
    "consumer_secret": "YChXvmHrSUj6CQybs2jIClH8L9bJ1QkOfRZ87ai0nIWMRX3OiT",
    "oauth_version": "1.0",
    "oauth_signature": "HMAC-SHA1",
    "oauth_callback": "http://eff47623.ngrok.io/request-token"
}
```



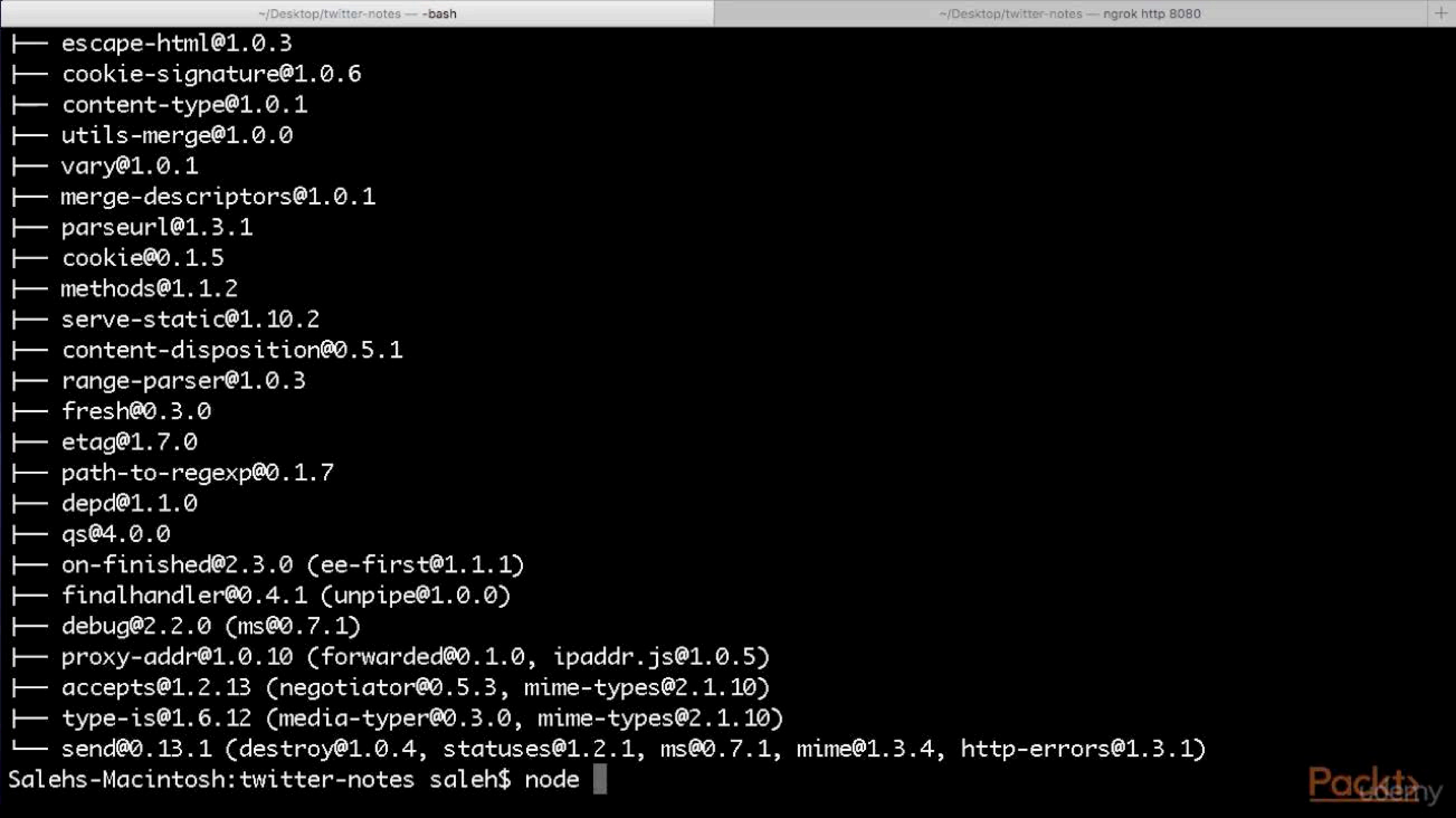
```
var OAuth = require('oauth').OAuth;
var config = require('./config');

// Create the oauth object for accessing Twitter
var oauth = new OAuth(
    config.request_token_url,
    config.access_token_url,
    config.consumer_key,
    config.consumer_secret,
    config.oauth_version,
    config.oauth_callback,
    config.oauth_signature
);

module.exports = {
    redirectToTwitterLoginPage: function(req, res) {
        // Ask Twitter for a request token
        oauth.getOAuthRequestToken(function(error, oauth_token, oauth_token_secret,
            if (error) {
                console.log(error);
                res.send("Authentication failed!");
            } else {
                // Use the request token to take the client to Twitter's authentica
                res.cookie('oauth_token', oauth_token, { httpOnly: true });
                res.cookie('oauth_token_secret', oauth_token_secret, { httpOnly: tru
                res.redirect(config.authorize_url + '?oauth_token='+oauth_token);
            }
        });
```



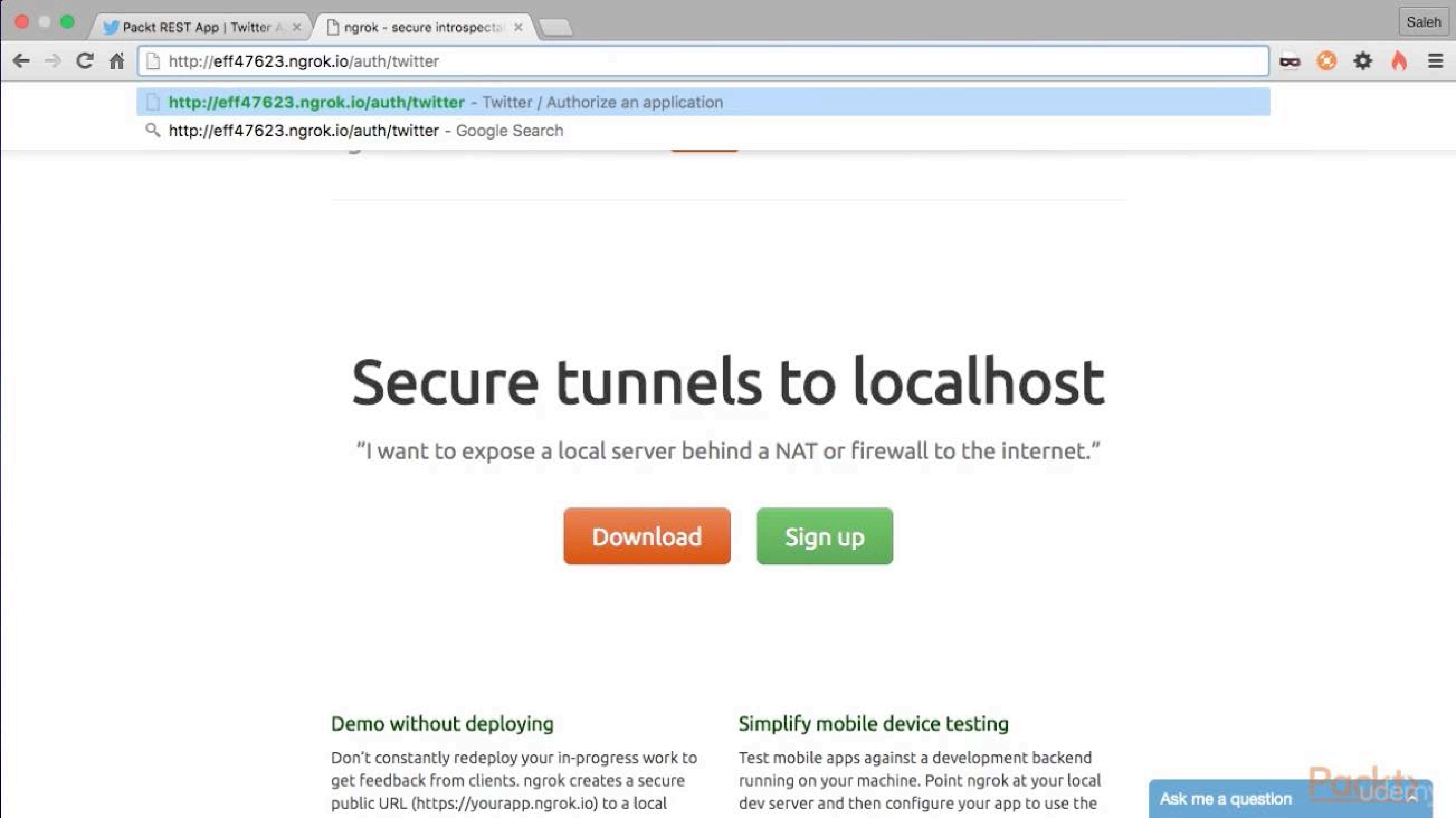
```javascript
module.exports = {
    redirectToTwitterLoginPage: function(req, res) {
        // Ask Twitter for a request token
        oauth.getOAuthRequestToken(function(error, oauth_token, oauth_token_secret, 
            if (error) {
                console.log(error);
                res.send("Authentication failed!");
            } else {
                // Use the request token to take the client to Twitter's authentica
                res.cookie('oauth_token', oauth_token, { httpOnly: true });
                res.cookie('oauth_token_secret', oauth_token_secret, { httpOnly: tru
                res.redirect(config.authorize_url + '?oauth_token='+oauth_token);
            }
        });
    },
    authenticate: function(req, res, cb) {
        // Check if the request token and temporary credential are there
        if (!(req.cookies.oauth_token && req.cookies.oauth_token_secret && req.query
            return cb("Request does not have all required keys");
        }

        // Clear the request token cookies
        res.clearCookie('oauth_token');
        res.clearCookie('oauth_token_secret');

        // Tell router that authentication was successful
        cb();
    }
```



```

LoginPage: function(req, res) {
r for a request token
hRequestToken(function(error, oauth_token, oauth_token_secret, results) {
) {
le.log(error);
end("Authentication failed!");

e the request token to take the client to Twitter's authentication page
ookie('oauth_token', oauth_token, { httpOnly: true });
ookie('oauth_token_secret', oauth_token_secret, { httpOnly: true });
edirect(config.authorize_url + '?oauth_token='+oauth_token);



ction(req, res, cb) {
he request token and temporary credential are there
kies.oauth_token && req.cookies.oauth_token_secret && req.query.oauth_verifier)) {
("Request does not have all required keys");


request token cookies
ie('oauth_token');
ie('oauth_token_secret');

r that authentication was successful

```


~/Desktop/twitter-notes — -bash
~/Desktop/twitter-notes — ngrok http 8080

```
├── escape-html@1.0.3
├── cookie-signature@1.0.6
├── content-type@1.0.1
├── utils-merge@1.0.0
├── vary@1.0.1
├── merge-descriptors@1.0.1
├── parseurl@1.3.1
├── cookie@0.1.5
├── methods@1.1.2
├── serve-static@1.10.2
├── content-disposition@0.5.1
├── range-parser@1.0.3
├── fresh@0.3.0
├── etag@1.7.0
├── path-to-regexp@0.1.7
├── depd@1.1.0
├── qs@4.0.0
├── on-finished@2.3.0 (ee-first@1.1.1)
├── finalhandler@0.4.1 (unpipe@1.0.0)
├── debug@2.2.0 (ms@0.7.1)
├── proxy-addr@1.0.10 (forwarded@0.1.0, ipaddr.js@1.0.5)
├── accepts@1.2.13 (negotiator@0.5.3, mime-types@2.1.10)
├── type-is@1.6.12 (media-typer@0.3.0, mime-types@2.1.10)
└── send@0.13.1 (destroy@1.0.4, statuses@1.2.1, ms@0.7.1, mime@1.3.4, http-errors@1.3.1)
Salehs-Macintosh:twitter-notes saleh$ node
```


Packt REST App | Twitter A
ngrok - secure introspecta
Saleh
http://eff47623.ngrok.io/auth/twitter
http://eff47623.ngrok.io/auth/twitter - Twitter / Authorize an application
http://eff47623.ngrok.io/auth/twitter - Google Search

Secure tunnels to localhost

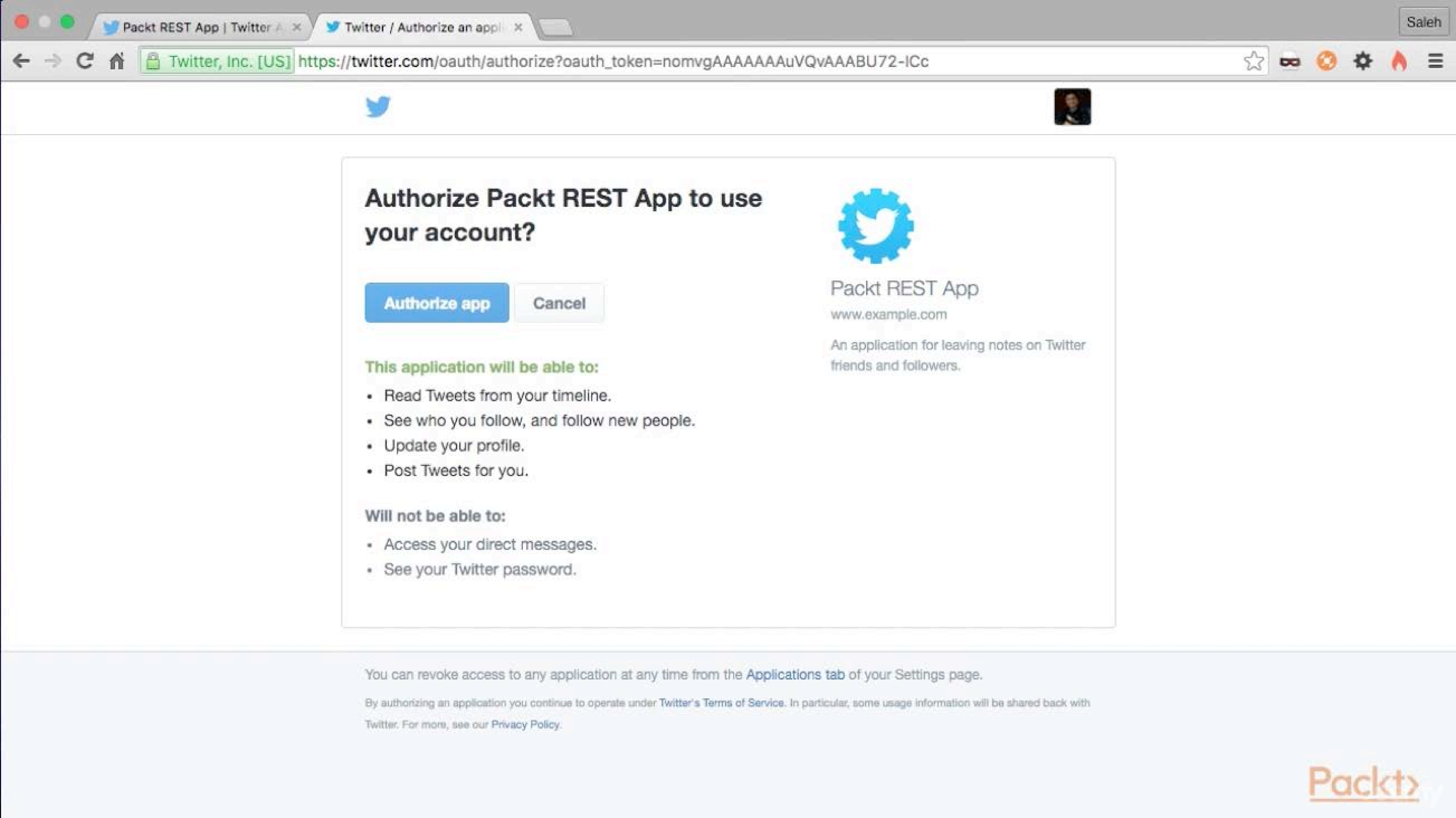
"I want to expose a local server behind a NAT or firewall to the internet."

Demo without deploying

Don't constantly redeploy your in-progress work to get feedback from clients. ngrok creates a secure public URL (https://yourapp.ngrok.io) to a local

Simplify mobile device testing

Test mobile apps against a development backend running on your machine. Point ngrok at your local dev server and then configure your app to use the


Packt REST App | Twitter A
Twitter / Authorize an appl
Saleh
Twitter, Inc. [US]
https://twitter.com/oauth/authorize?oauth_token=nomvgAAAAAAAuVQvAAABU72-ICc
Authorize app
Cancel

Authorize Packt REST App to use your account?

This application will be able to:

- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.

Will not be able to:

- Access your direct messages.
- See your Twitter password.

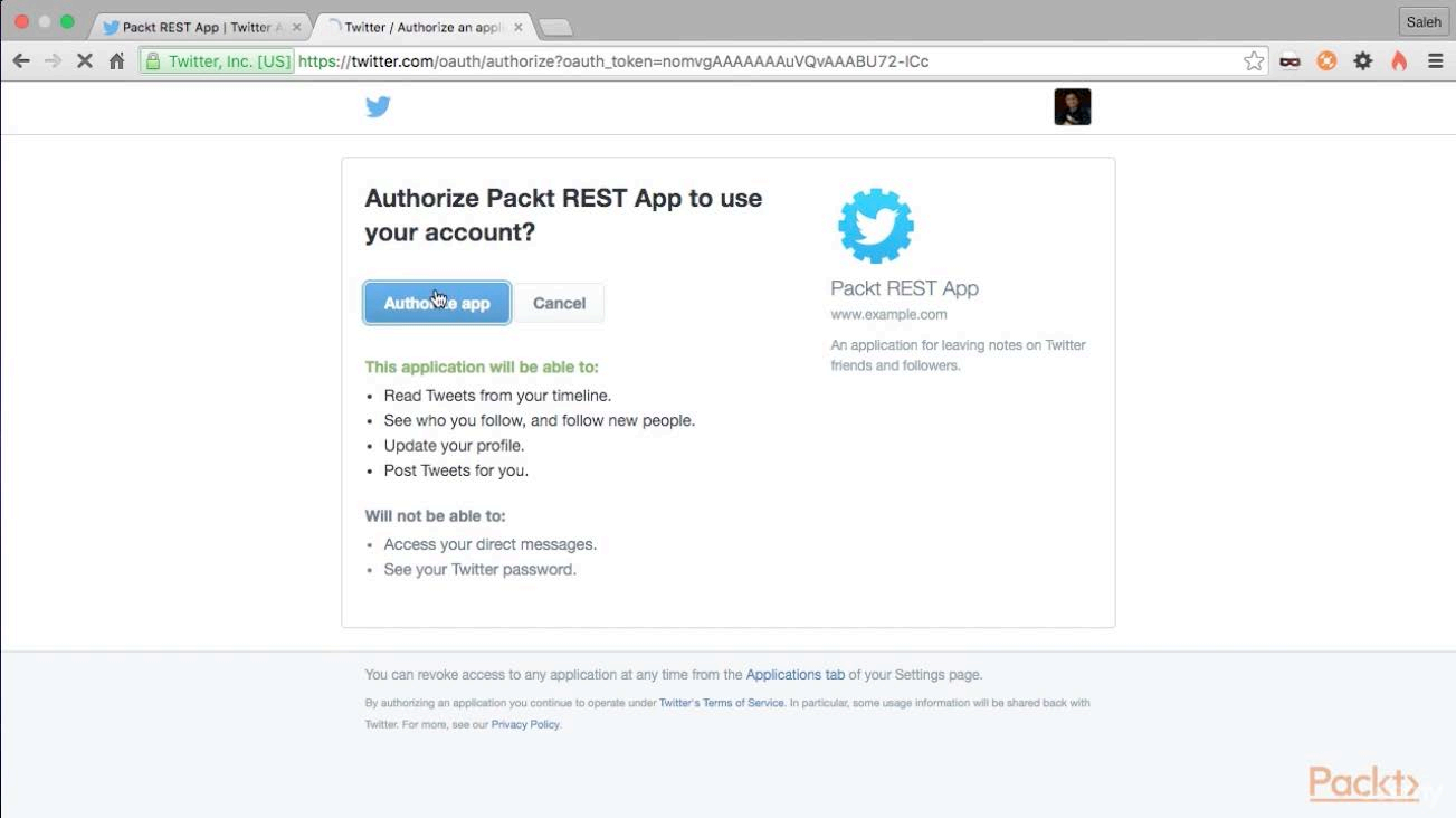
Packt REST App

www.example.com

An application for leaving notes on Twitter friends and followers.

You can revoke access to any application at any time from the Applications tab of your Settings page.

By authorizing an application you continue to operate under Twitter's Terms of Service. In particular, some usage information will be shared back with Twitter. For more, see our Privacy Policy.


Packt REST App | Twitter A
Twitter / Authorize an appl
Saleh
Twitter, Inc. [US]
https://twitter.com/oauth/authorize?oauth_token=nomvgAAAAAAAuVQvAAABU72-ICc
Authorize app
Cancel

Authorize Packt REST App to use your account?

This application will be able to:

- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.

Will not be able to:

- Access your direct messages.
- See your Twitter password.

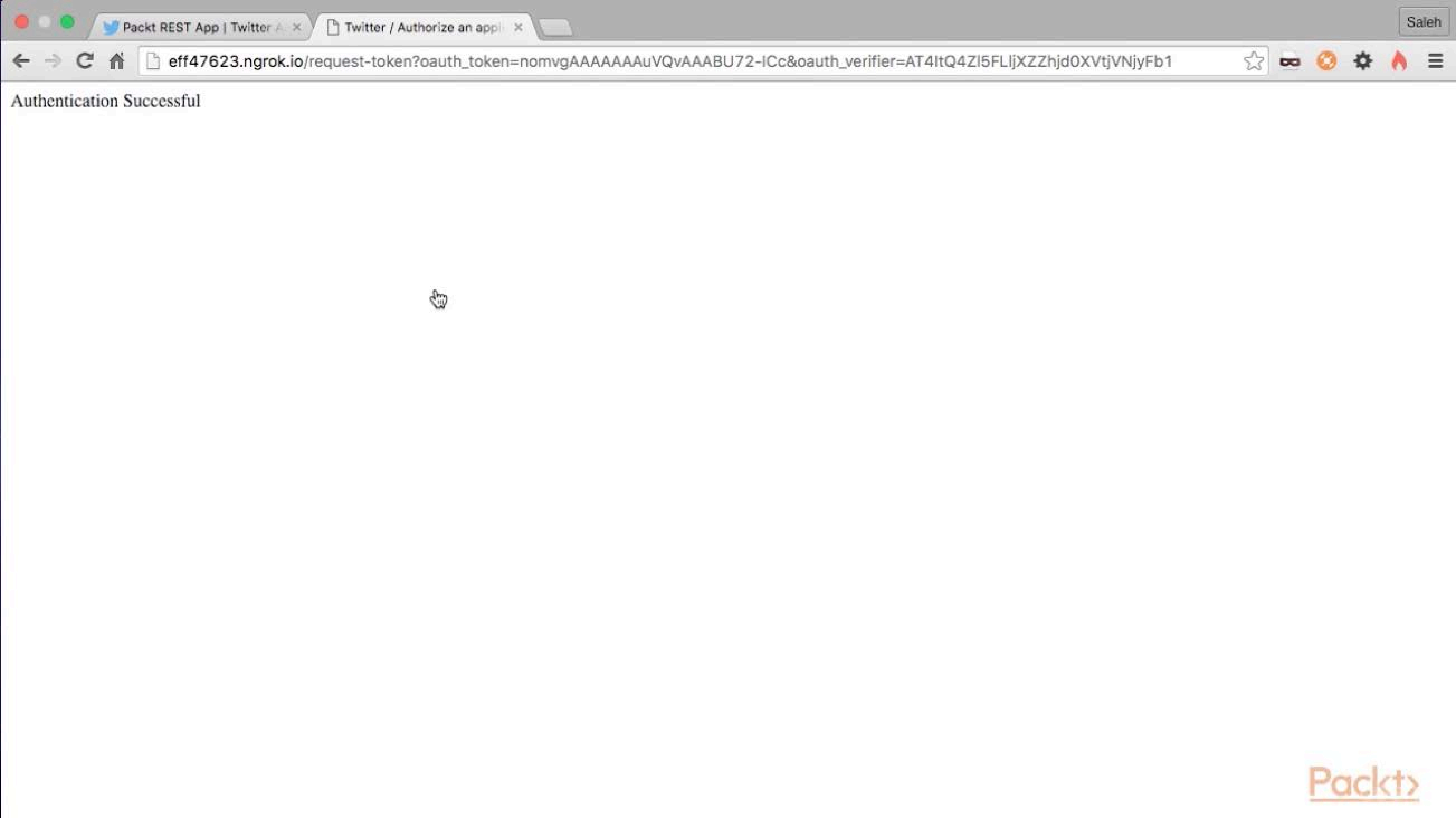
Packt REST App

www.example.com

An application for leaving notes on Twitter friends and followers.

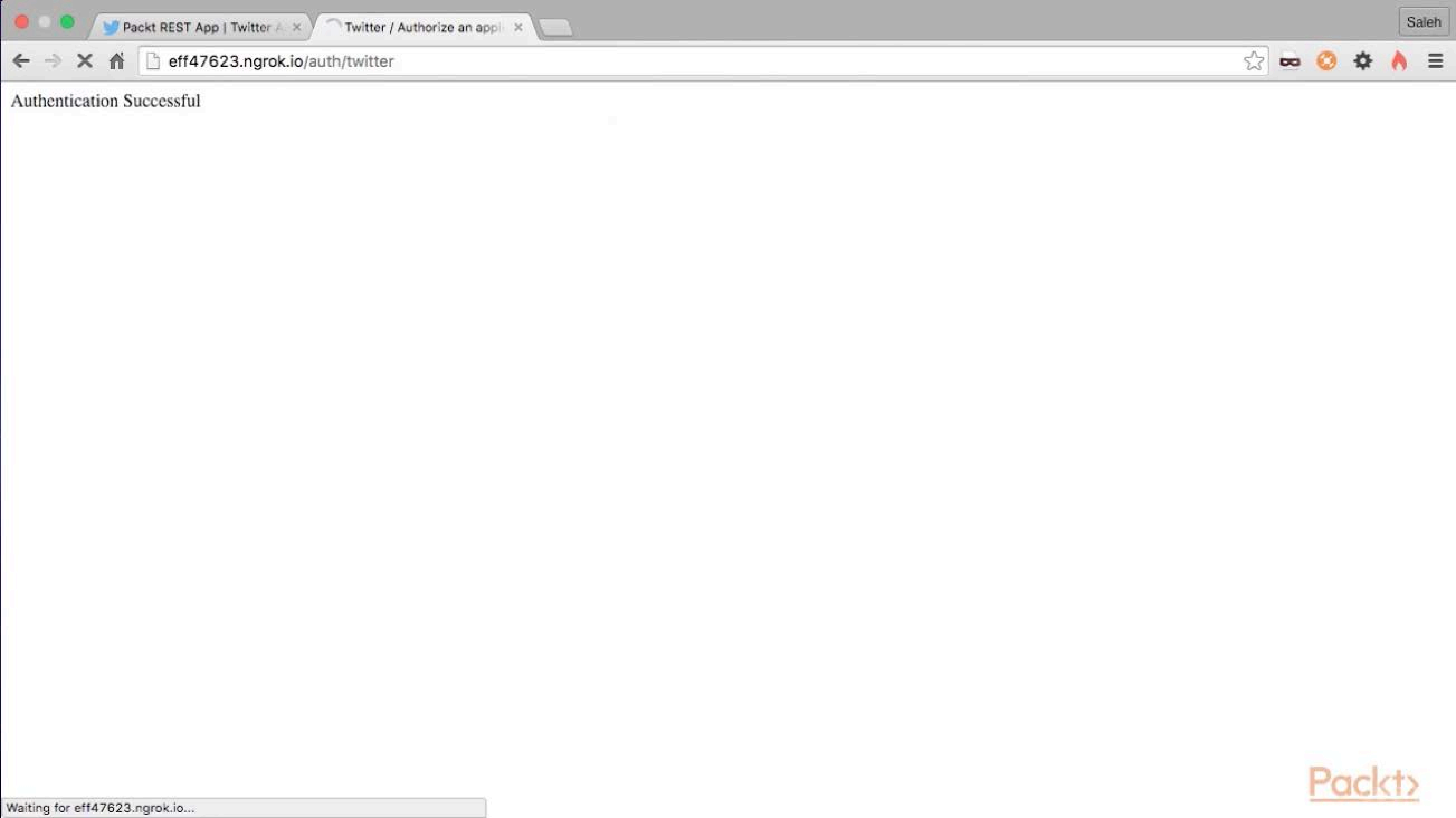
You can revoke access to any application at any time from the Applications tab of your Settings page.

By authorizing an application you continue to operate under Twitter's Terms of Service. In particular, some usage information will be shared back with Twitter. For more, see our Privacy Policy.



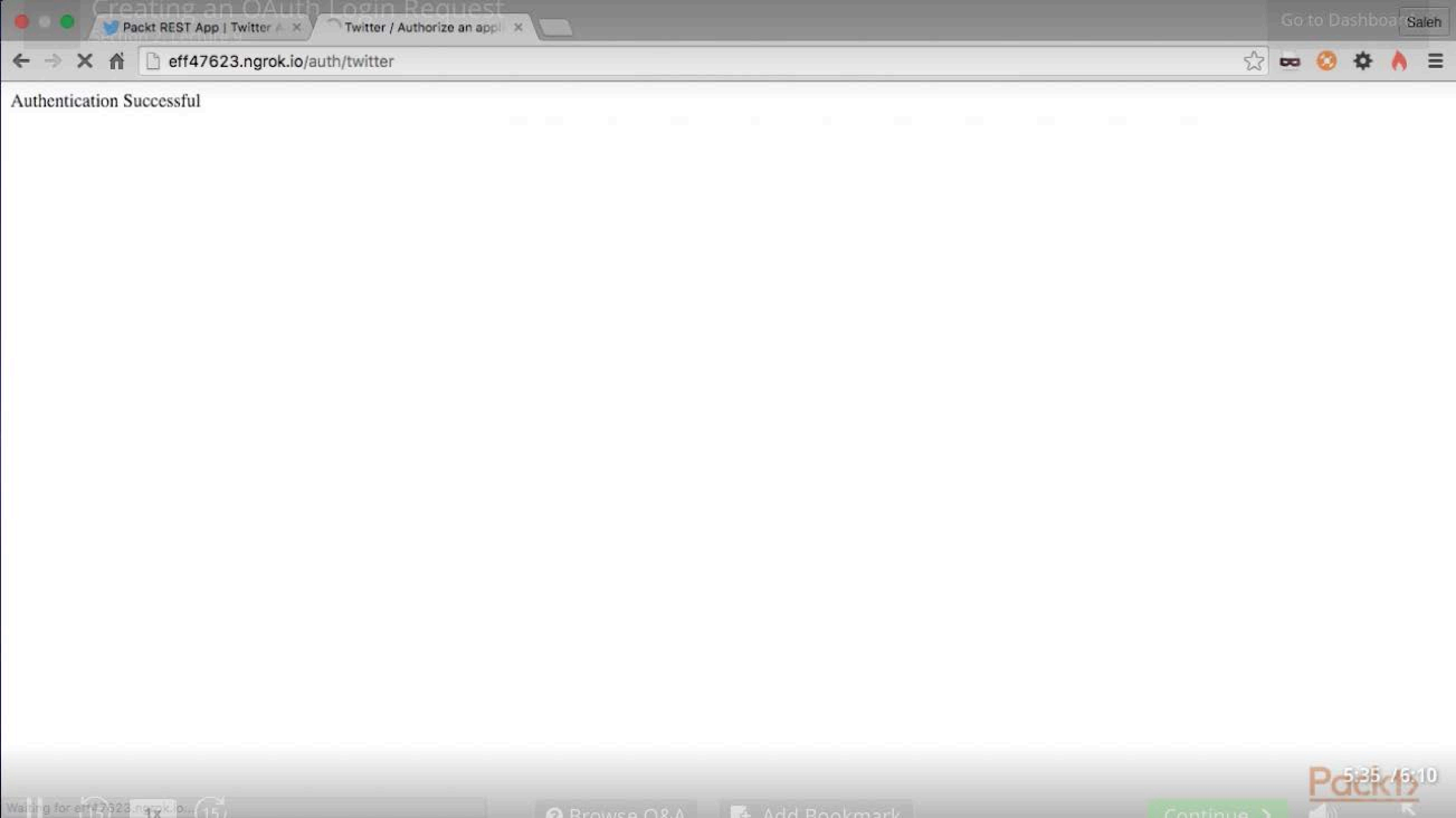
eff47623.ngrok.io/request-token?oauth_token=nomvgAAAAAAuVQvAAABU72-ICc&oauth_verifier=AT4ItQ4ZI5FLljXZZhjd0XVtjVNjyFb1

Authentication Successful


Packt REST App | Twitter A
Twitter / Authorize an appl
Saleh
eff47623.ngrok.io/auth/twitter

Authentication Successful

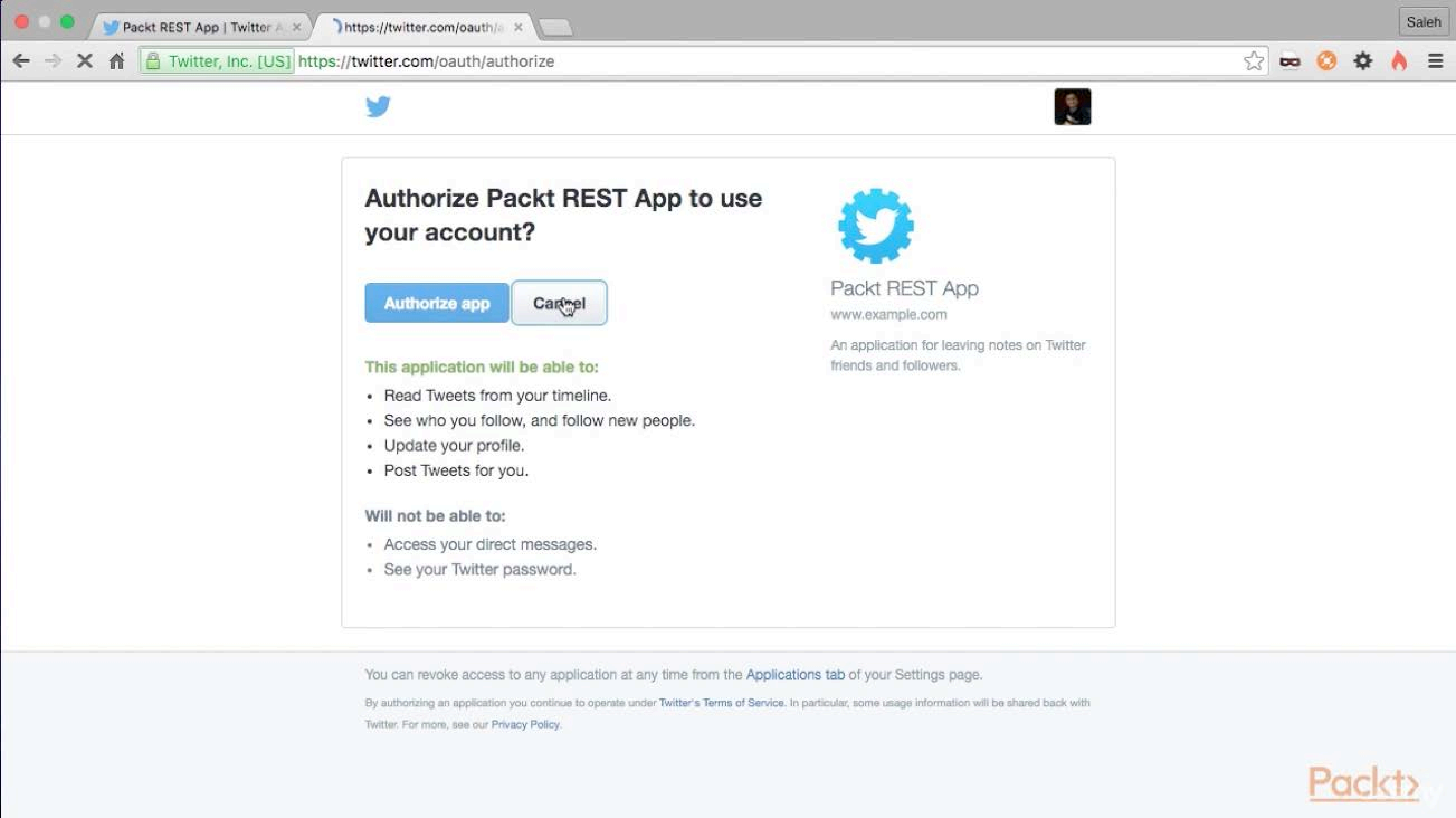
Waiting for eff47623.ngrok.io...
Packt>



Creating an OAuth Login Request
Packt REST App | Twitter A
Twitter / Authorize an appl
Go to Dashboard
Saleh
eff47623.ngrok.io/auth/twitter

Authentication Successful

Waiting for eff47623.ngrok.io...
1x
15
Browse Q&A
Add Bookmark
Continue
5:35 / 6:10
Packt



Packt REST App | Twitter A
https://twitter.com/oauth/a
Saleh
Twitter, Inc. [US]
https://twitter.com/oauth/authorize

Authorize Packt REST App to use your account?

Packt REST App

www.example.com

An application for leaving notes on Twitter friends and followers.

This application will be able to:

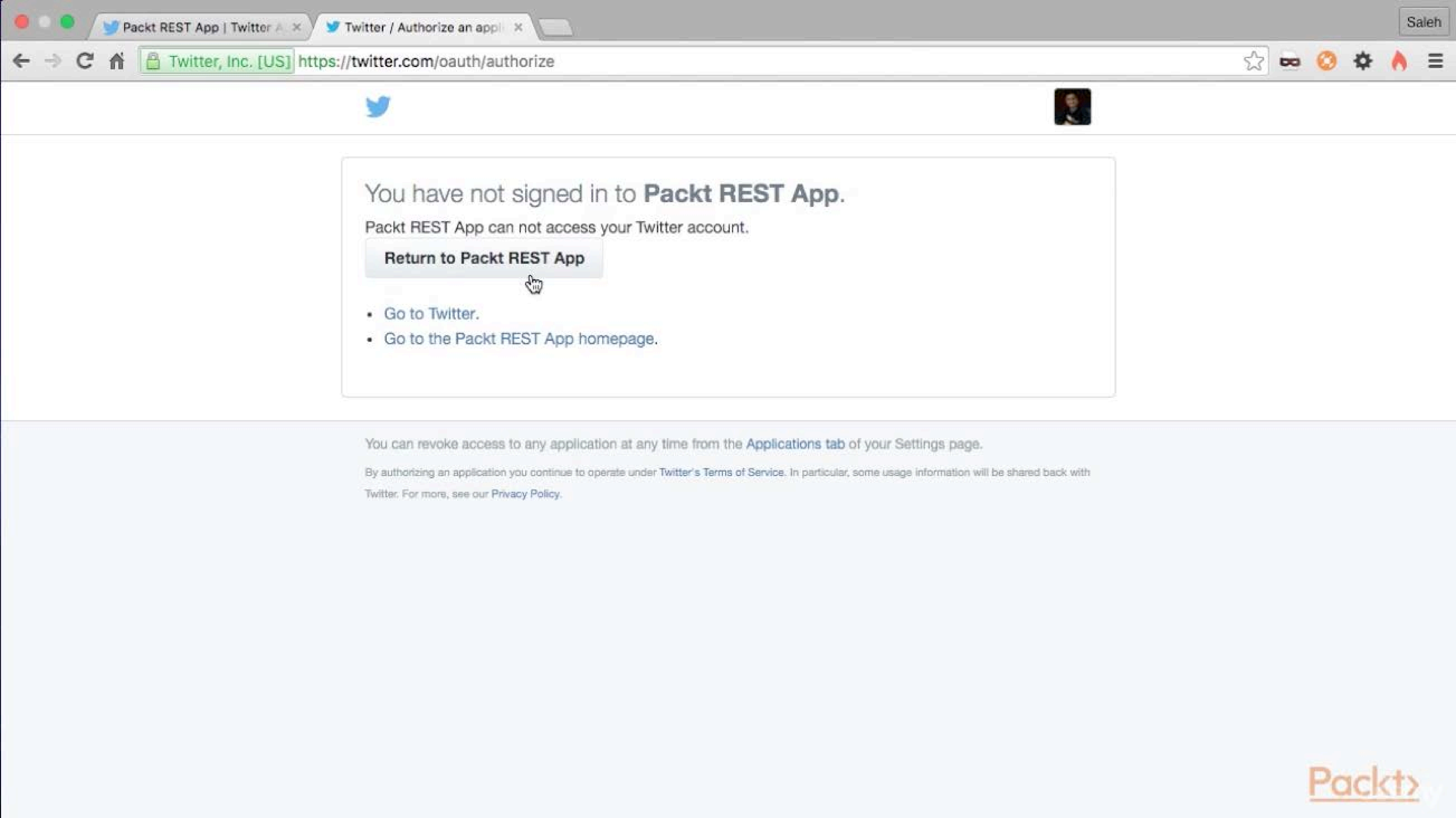
- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.

Will not be able to:

- Access your direct messages.
- See your Twitter password.

You can revoke access to any application at any time from the Applications tab of your Settings page.

By authorizing an application you continue to operate under Twitter's Terms of Service. In particular, some usage information will be shared back with Twitter. For more, see our Privacy Policy.


Packt REST App | Twitter A
Twitter / Authorize an appl
Saleh
Twitter, Inc. [US]
https://twitter.com/oauth/authorize

You have not signed in to **Packt REST App**.

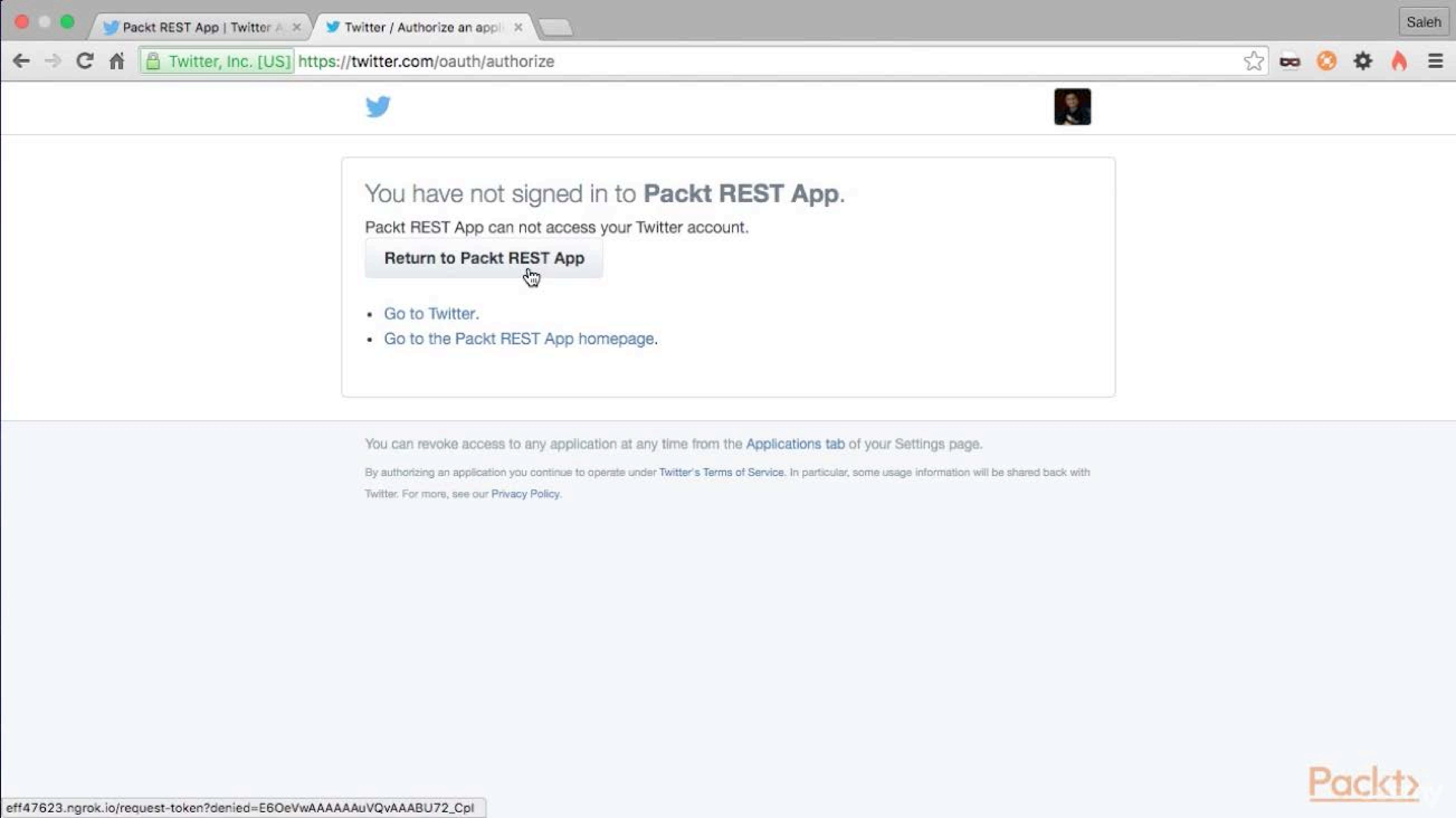
Packt REST App can not access your Twitter account.

Return to Packt REST App

- Go to Twitter.
- Go to the Packt REST App homepage.

You can revoke access to any application at any time from the Applications tab of your Settings page.

By authorizing an application you continue to operate under Twitter's Terms of Service. In particular, some usage information will be shared back with Twitter. For more, see our Privacy Policy.


Packt REST App | Twitter A
Twitter / Authorize an appl
Saleh
Twitter, Inc. [US]
https://twitter.com/oauth/authorize

You have not signed in to **Packt REST App**.

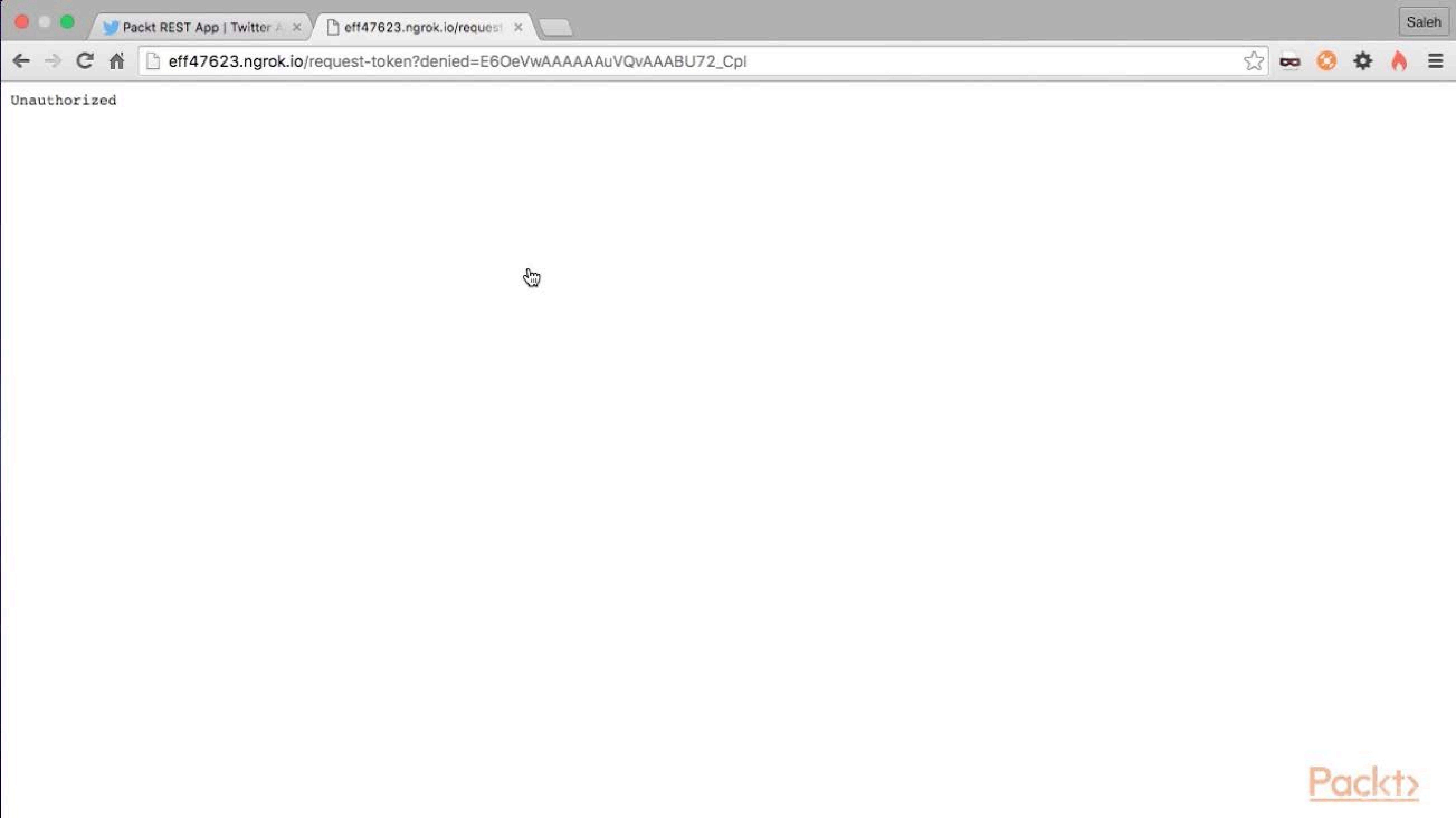
Packt REST App can not access your Twitter account.

- Go to Twitter.
- Go to the Packt REST App homepage.

You can revoke access to any application at any time from the Applications tab of your Settings page.

By authorizing an application you continue to operate under Twitter's Terms of Service. In particular, some usage information will be shared back with Twitter. For more, see our Privacy Policy.

```
eff47623.ngrok.io/request-token?denied=E6OeVwAAAAAAuVQvAAABU72_Cpl
```



Packt REST App | Twitter A
eff47623.ngrok.io/request
Saleh
eff47623.ngrok.io/request-token?denied=E6OeVwAAAAAAuVQvAAABU72_Cpl

```
Unauthorized
```

Next Video

Getting an Access Token