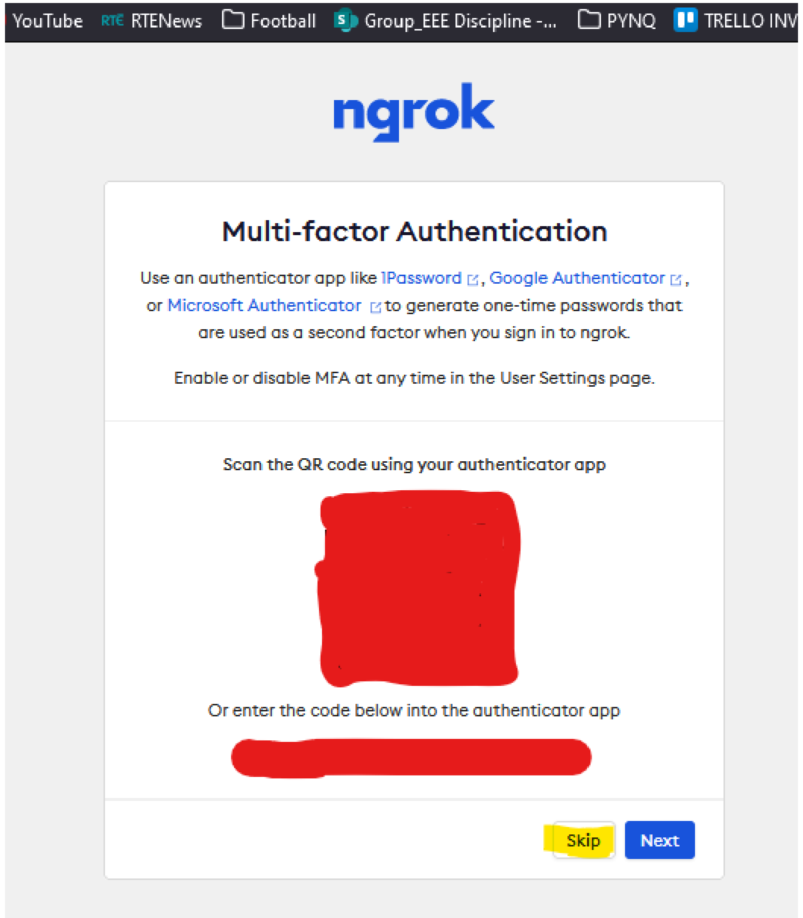
Creating and Configuring an Account at ngrok.com

Create account – Click link sent by email- Skip 2FA



Multi-factor Authentication

Use an authenticator app like 1Password, Google Authenticator, or Microsoft Authenticator to generate one-time passwords that are used as a second factor when you sign in to ngrok.

Enable or disable MFA at any time in the User Settings page.

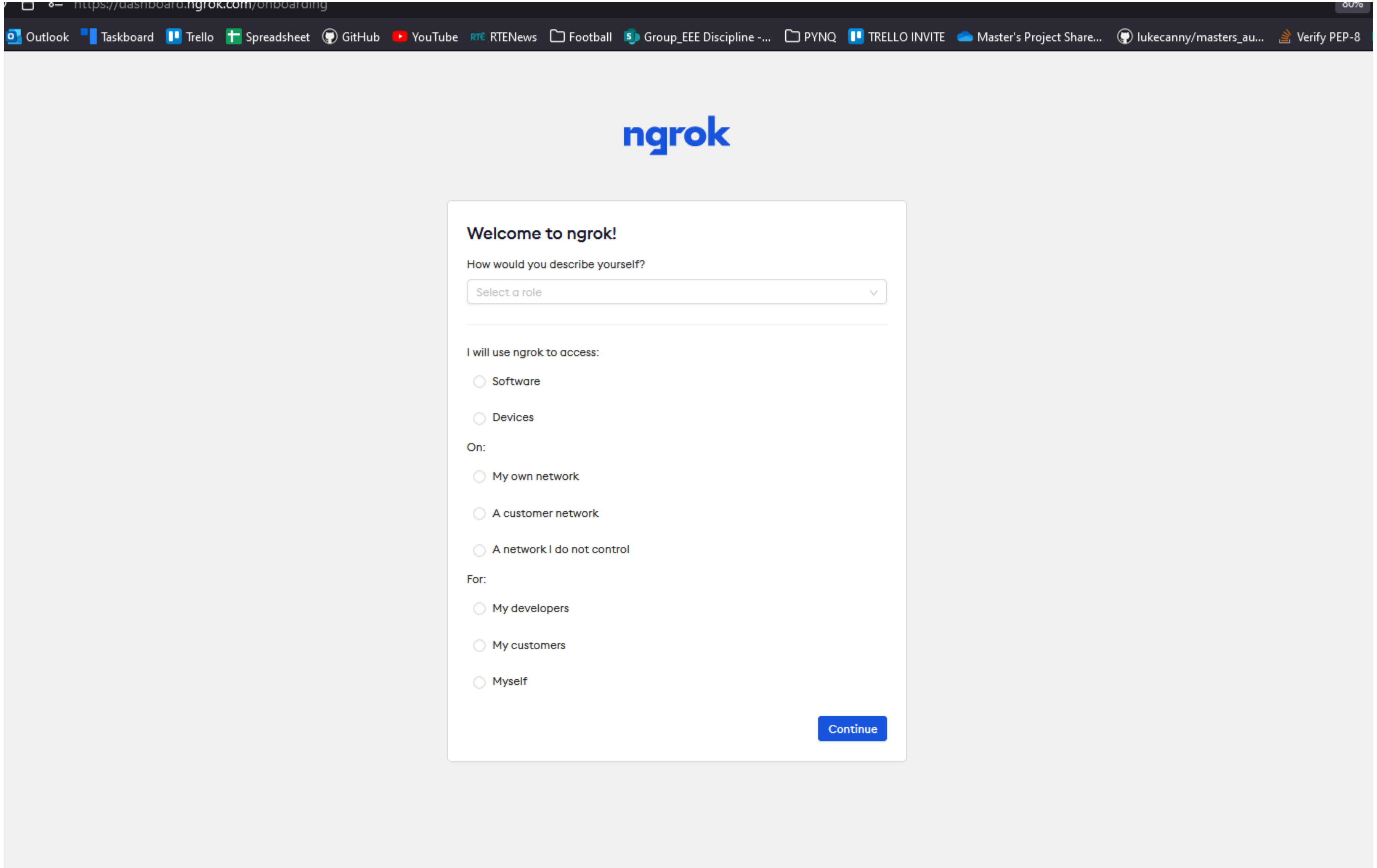
Scan the QR code using your authenticator app

[redacted]

Or enter the code below into the authenticator app

[redacted]

Doesn’t need to be filled out, just continue



Welcome to ngrok!

How would you describe yourself?

Select a role

I will use ngrok to access:

○ Software

○ Devices

On:

○ My own network

○ A customer network

○ A network I do not control

For:

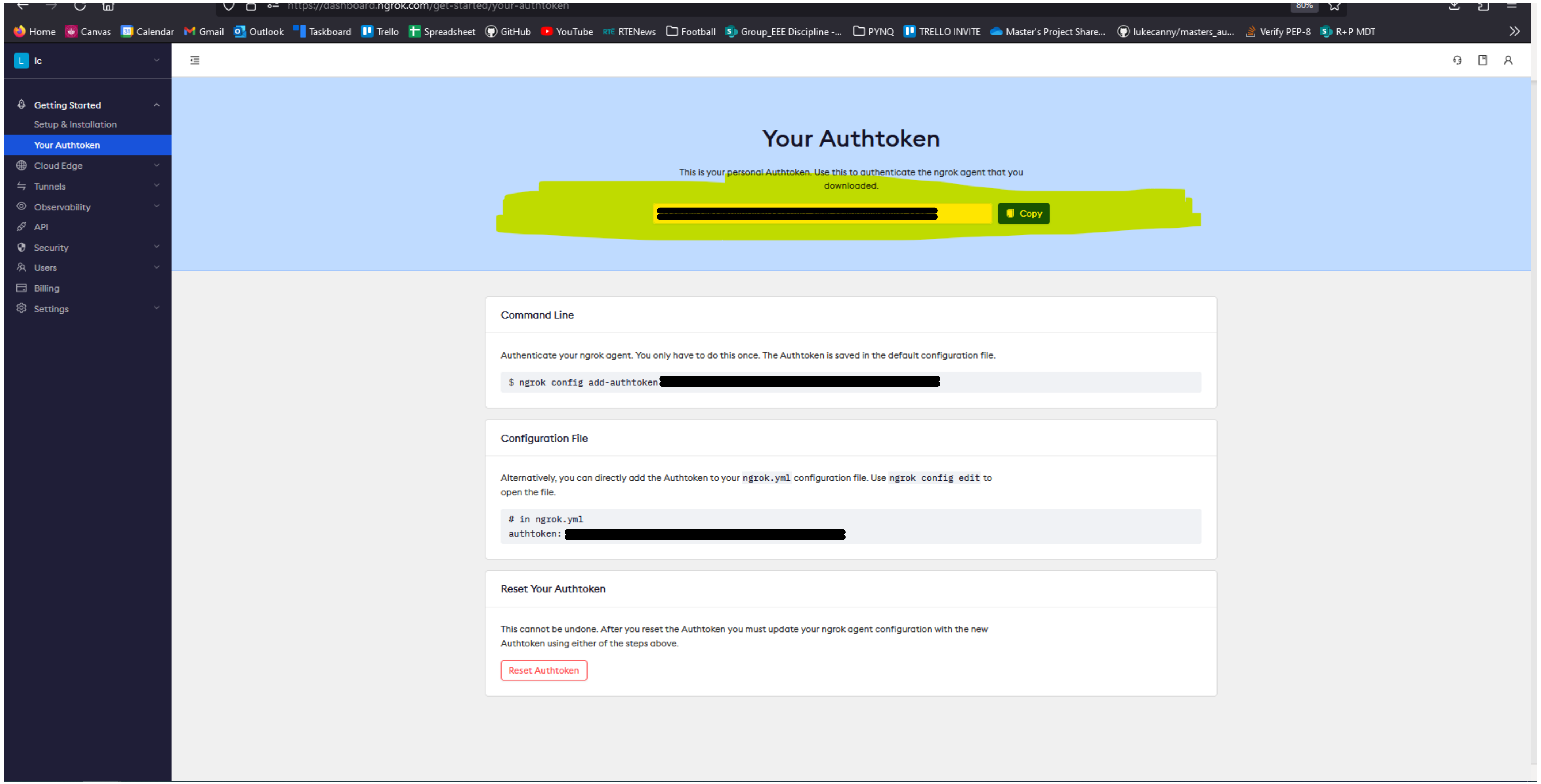
○ My developers

○ My customers

○ Myself

Continue

Click on “Your Authtoken” on top left and make note of auth token



Your Authtoken

This is your personal Authtoken. Use this to authenticate the ngrok agent that you downloaded.

[redacted]

Command Line

Authenticate your ngrok agent. You only have to do this once. The Authtoken is saved in the default configuration file.

```
$ ngrok config add-authtoken [redacted]
```

Configuration File

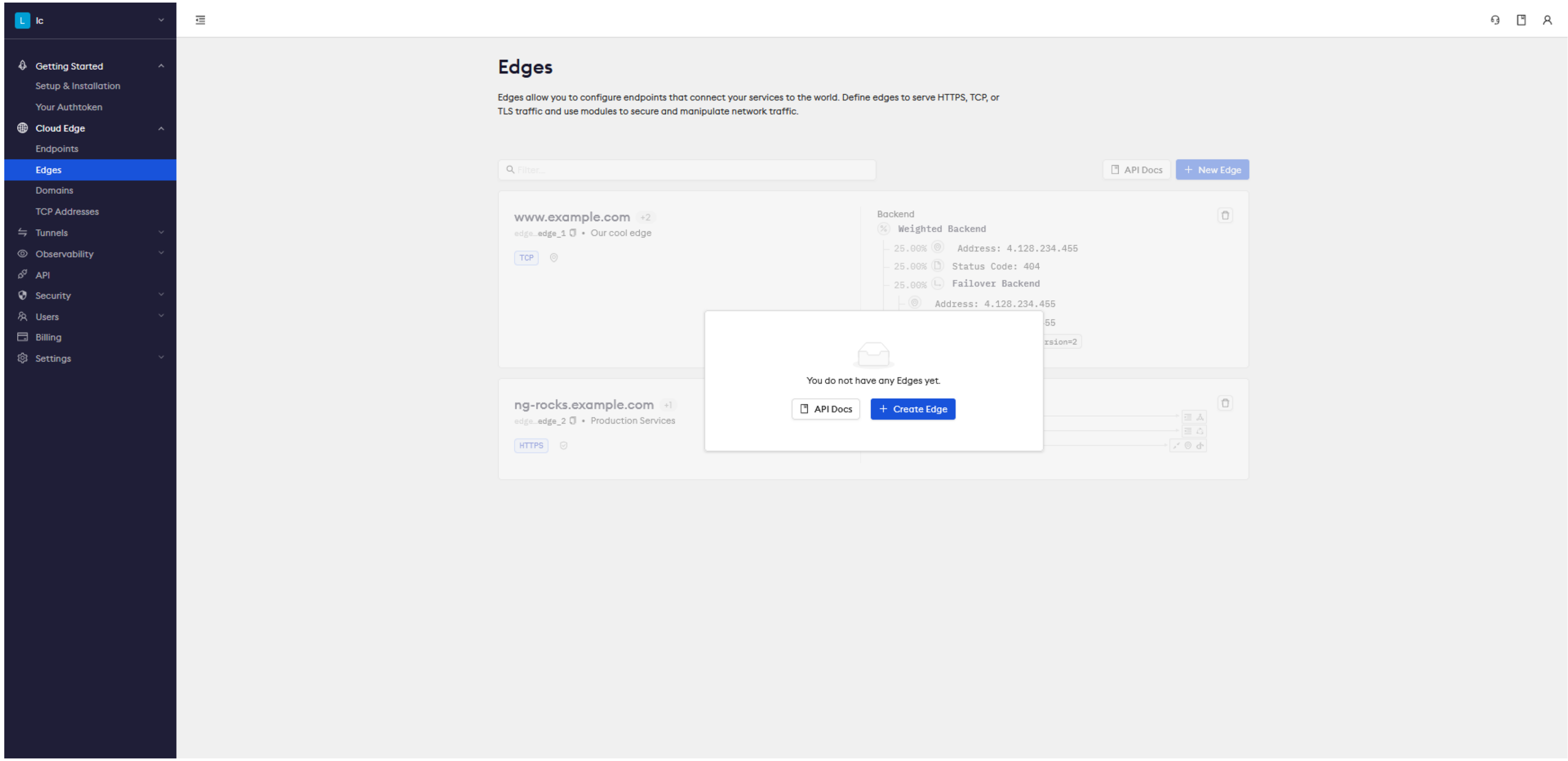
Alternatively, you can directly add the Authtoken to your `ngrok.yml` configuration file. Use `ngrok config edit` to open the file.

```
# in ngrok.yml
authtoken: [redacted]
```

Reset Your Authtoken

This cannot be undone. After you reset the Authtoken you must update your ngrok agent configuration with the new Authtoken using either of the steps above.

Next under “Cloud Edge” select “Edges” – Click “Create Edge”

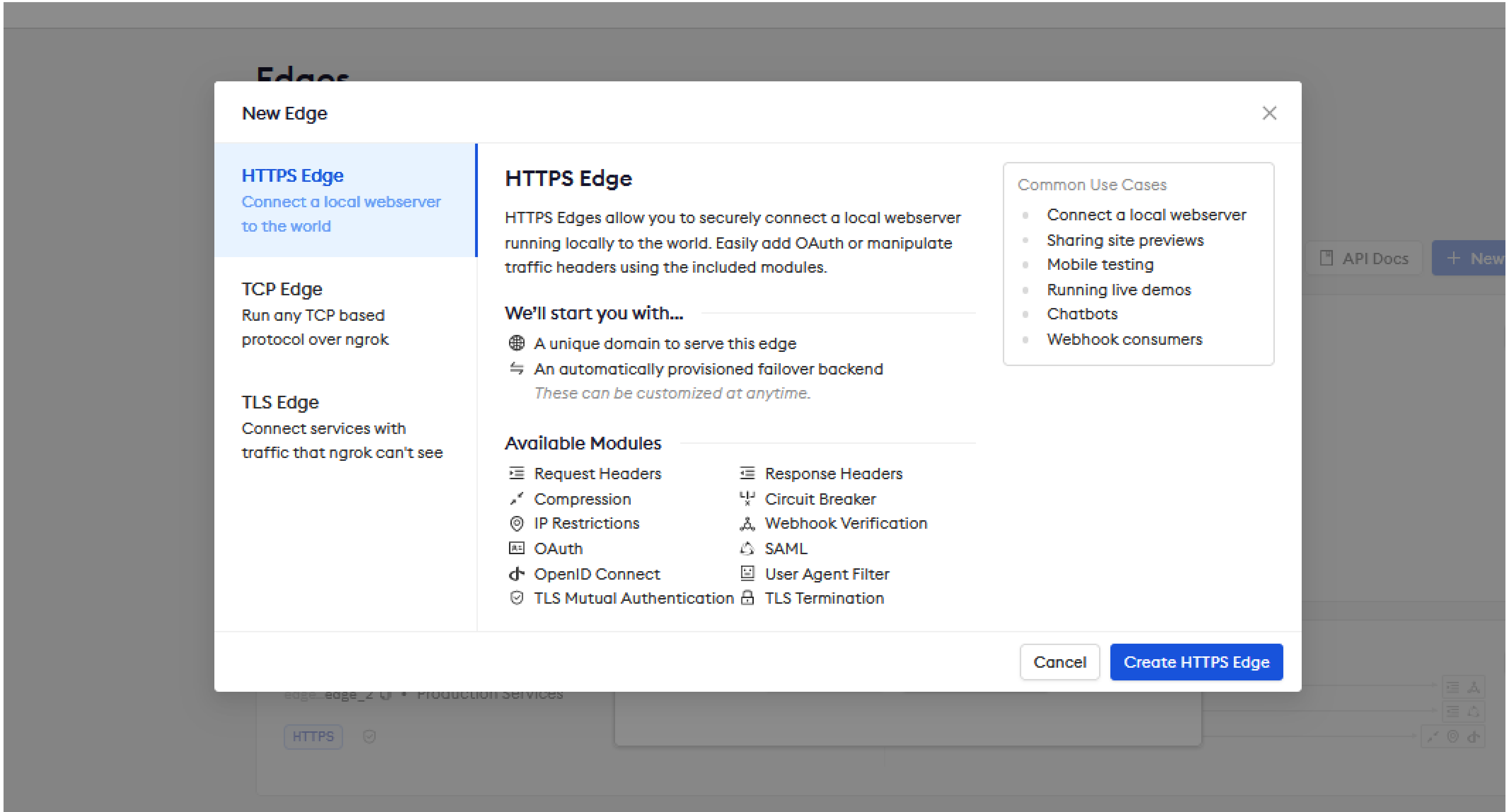


Edges

Edges allow you to configure endpoints that connect your services to the world. Define edges to serve HTTPS, TCP, or TLS traffic and use modules to secure and manipulate network traffic.

You do not have any Edges yet.

Choose HTTPS Edge



New Edge

HTTPS Edge
Connect a local webserver to the world

TCP Edge
Run any TCP based protocol over ngrok

TLS Edge
Connect services with traffic that ngrok can't see

HTTPS Edge

HTTPS Edges allow you to securely connect a local webserver running locally to the world. Easily add OAuth or manipulate traffic headers using the included modules.

We'll start you with...

- A unique domain to serve this edge
- An automatically provisioned failover backend

These can be customized at anytime.

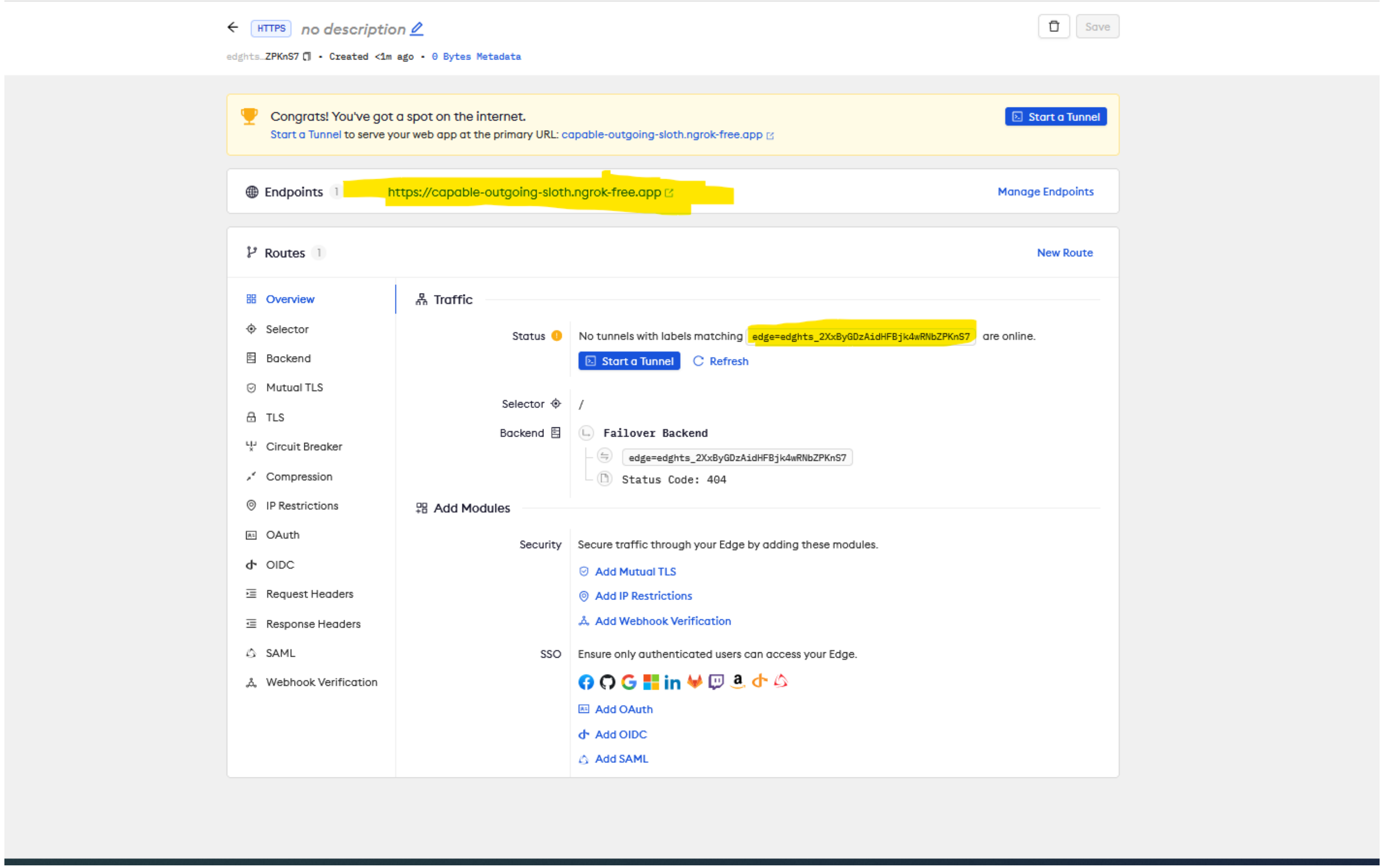
Available Modules

- Request Headers
- Compression
- IP Restrictions
- OAuth
- OpenID Connect
- TLS Mutual Authentication
- Response Headers
- Circuit Breaker
- Webhook Verification
- SAML
- User Agent Filter
- TLS Termination

Common Use Cases

- Connect a local webserver
- Sharing site previews
- Mobile testing
- Running live demos
- Chatbots
- Webhook consumers

Take note of the edge label AND the endpoint web address



edghts…ZPKnS7 • Created <1m ago • 0 Bytes Metadata

Congrats! You've got a spot on the internet.
Start a Tunnel to serve your web app at the primary URL: capable-outgoing-sloth.ngrok-free.app

Endpoints 1 https://capable-outgoing-sloth.ngrok-free.app

Routes 1

Traffic

Status: No tunnels with labels matching `edge=edghts_2XxByGDzAidHFBjk4wRNbZPKnS7` are online.

Start a Tunnel Refresh

Selector: /

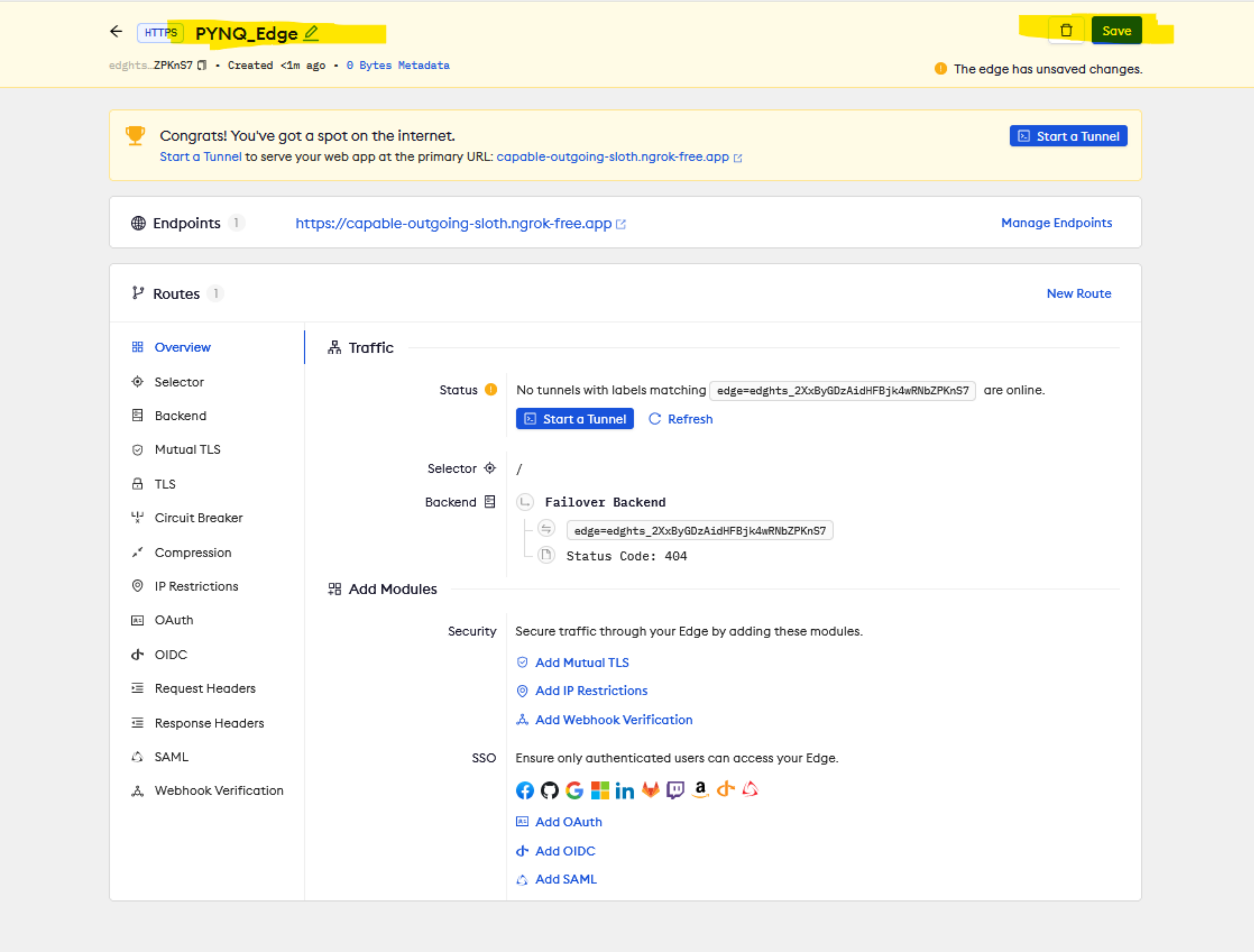
Backend: Failover Backend
`edge=edghts_2XxByGDzAidHFBjk4wRNbZPKnS7`
Status Code: 404

Add Modules

Security: Secure traffic through your Edge by adding these modules.
Add Mutual TLS
Add IP Restrictions
Add Webhook Verification

SSO: Ensure only authenticated users can access your Edge.
Add OAuth
Add OIDC
Add SAML

Rename edge to “PYNQ_Edge” and hit save



HTTPS PYNQ_Edge

edghts…ZPKnS7 • Created <1m ago • 0 Bytes Metadata

Save

The edge has unsaved changes.

Congrats! You've got a spot on the internet.
Start a Tunnel to serve your web app at the primary URL: capable-outgoing-sloth.ngrok-free.app

Start a Tunnel

Endpoints 1 https://capable-outgoing-sloth.ngrok-free.app Manage Endpoints

Routes 1 New Route

Traffic

Status: No tunnels with labels matching `edge=edghts_2XxByGDzAidHFBjk4wRNbZPKnS7` are online.

Start a Tunnel Refresh

Selector: /

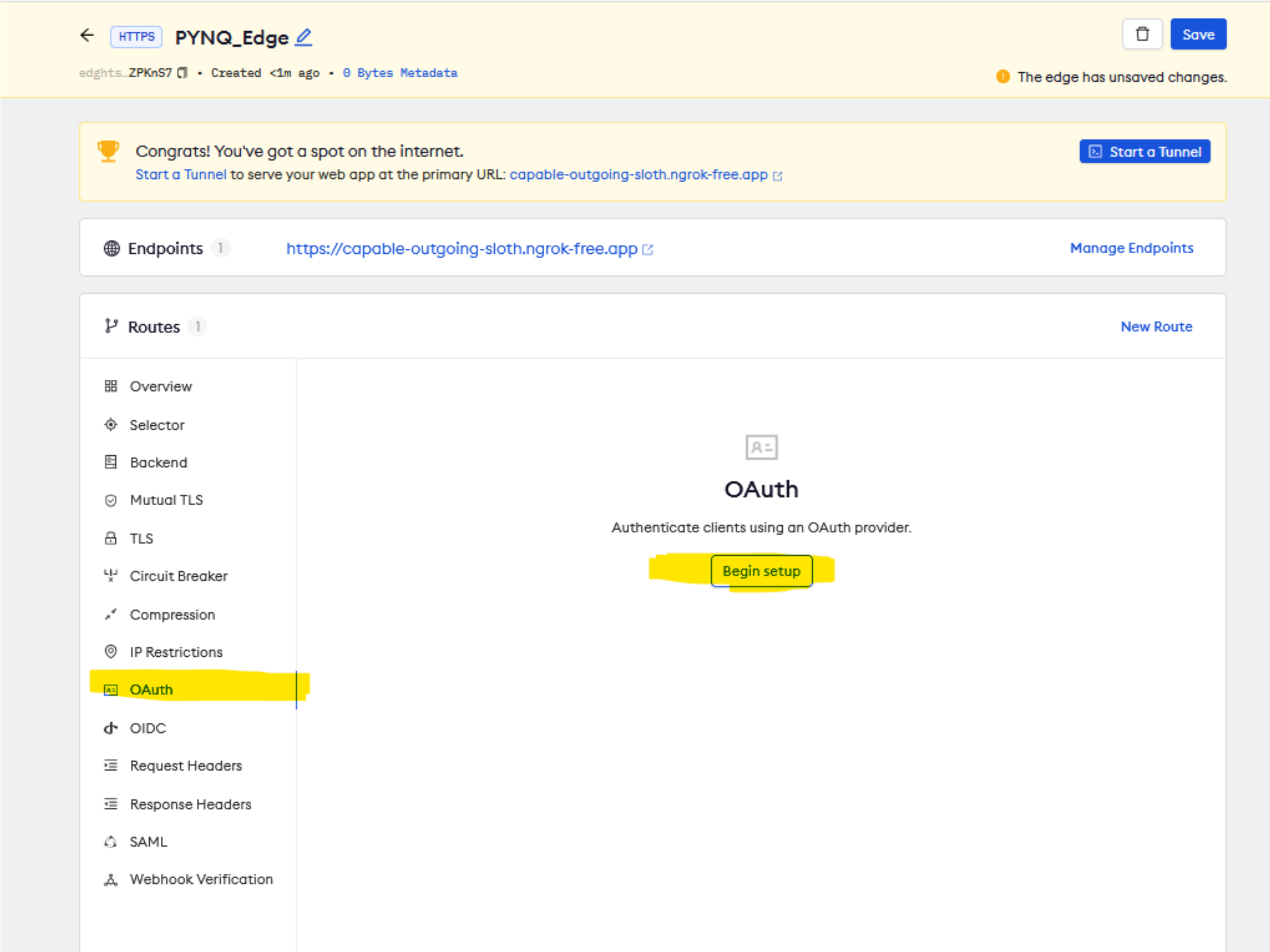
Backend: Failover Backend
`edge=edghts_2XxByGDzAidHFBjk4wRNbZPKnS7`
Status Code: 404

Add Modules

Security: Secure traffic through your Edge by adding these modules.
Add Mutual TLS
Add IP Restrictions
Add Webhook Verification

SSO: Ensure only authenticated users can access your Edge.
Add OAuth
Add OIDC
Add SAML

To set up Oauth – Click OAuth on LHS and click begin set up



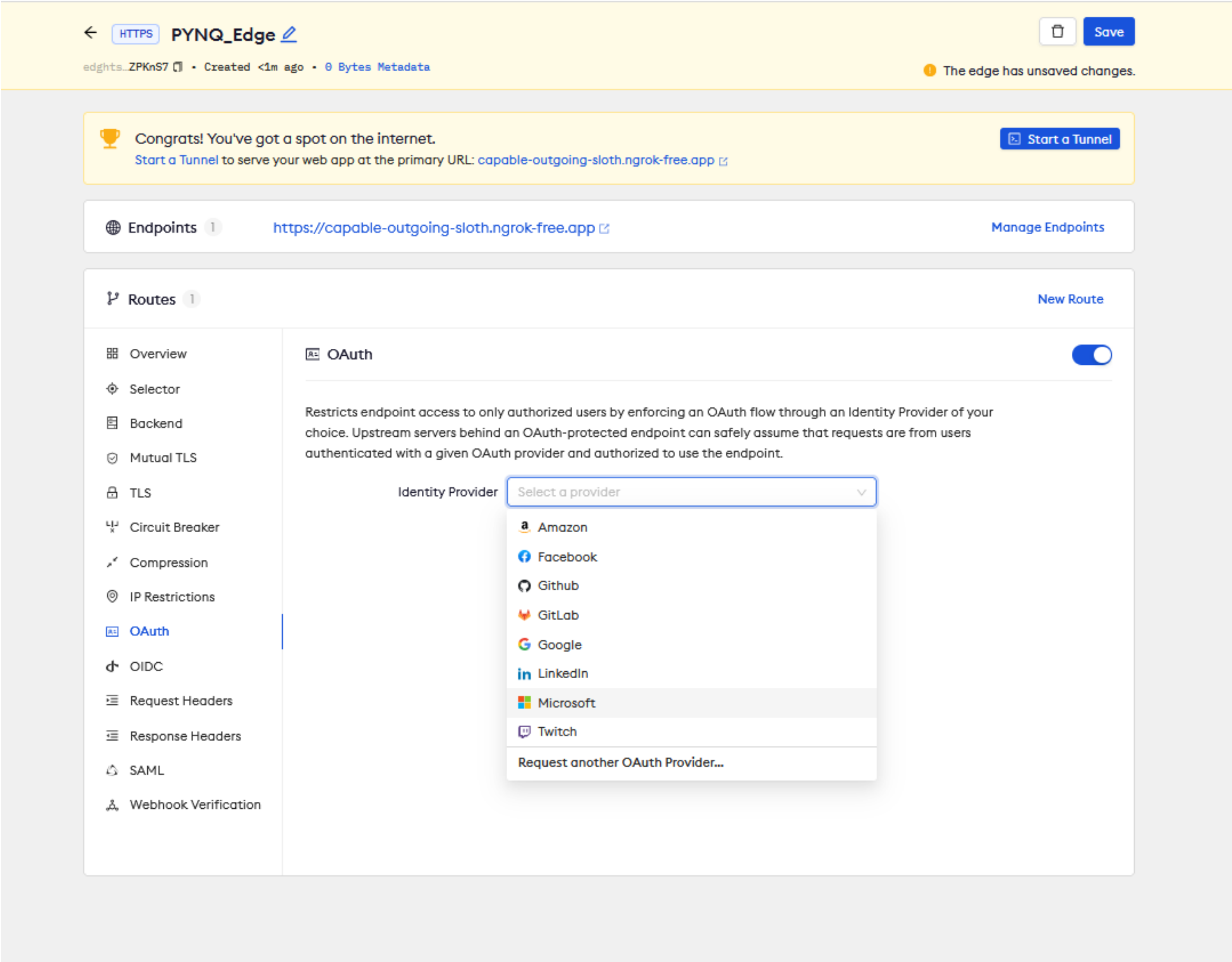
edghts…ZPKnS7 • Created <1m ago • 0 Bytes Metadata

The edge has unsaved changes.

Congrats! You've got a spot on the internet.
Start a Tunnel to serve your web app at the primary URL: capable-outgoing-sloth.ngrok-free.app

Authenticate clients using an OAuth provider.

Select the provider (I will choose Microsoft, although you can decide. This is how users will authenticate. Same as how Microsoft login used on Canvas/Blackboard)



HTTPS PYNQ_Edge

edghts_ZPKnS7 • Created <1m ago • 0 Bytes Metadata

Save

The edge has unsaved changes.

Congrats! You've got a spot on the internet.
Start a Tunnel to serve your web app at the primary URL: capable-outgoing-sloth.ngrok-free.app

Start a Tunnel

Endpoints 1 https://capable-outgoing-sloth.ngrok-free.app Manage Endpoints

Routes 1 New Route

- Overview
- Selector
- Backend
- Mutual TLS
- TLS
- Circuit Breaker
- Compression
- IP Restrictions
- OAuth
- OIDC
- Request Headers
- Response Headers
- SAML
- Webhook Verification

OAuth

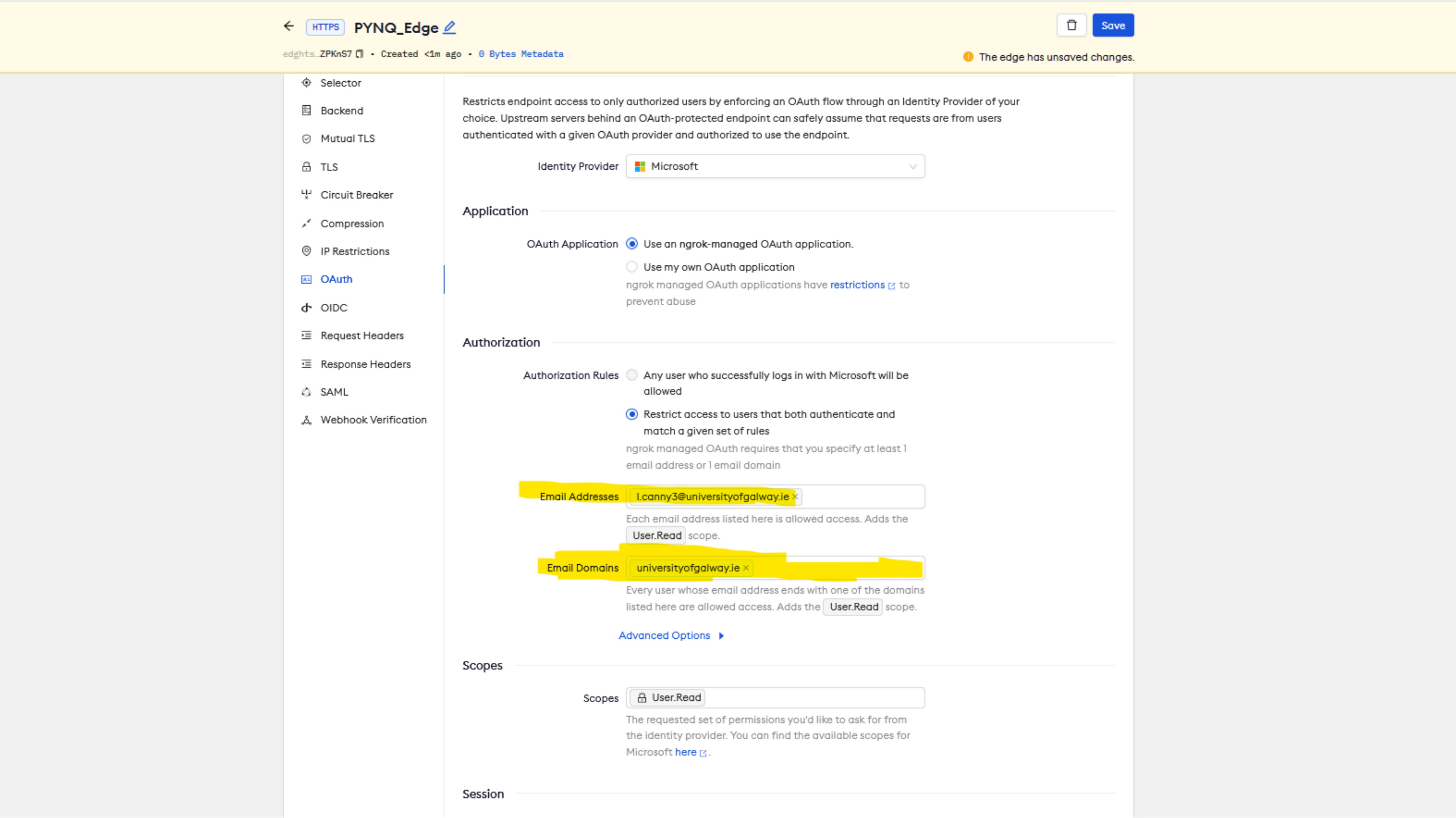
Restricts endpoint access to only authorized users by enforcing an OAuth flow through an Identity Provider of your choice. Upstream servers behind an OAuth-protected endpoint can safely assume that requests are from users authenticated with a given OAuth provider and authorized to use the endpoint.

Identity Provider: Select a provider

- Amazon
- Facebook
- Github
- GitLab
- Google
- LinkedIn
- Microsoft
- Twitch

Request another OAuth Provider...

Set either the specific email addresses OR email domain that is allowed to access your board. The college’s domain should suffice – i.e. only logins with @universityofgalway.ie are allowed. – remember to click save



PYNQ_Edge

edghts_ZPKnS7 • Created <1m ago • 0 Bytes Metadata

The edge has unsaved changes.

Restricts endpoint access to only authorized users by enforcing an OAuth flow through an Identity Provider of your choice. Upstream servers behind an OAuth-protected endpoint can safely assume that requests are from users authenticated with a given OAuth provider and authorized to use the endpoint.

Identity Provider: Microsoft

Application

OAuth Application
● Use an ngrok-managed OAuth application.
○ Use my own OAuth application
ngrok managed OAuth applications have restrictions to prevent abuse

Authorization

Authorization Rules
○ Any user who successfully logs in with Microsoft will be allowed
● Restrict access to users that both authenticate and match a given set of rules
ngrok managed OAuth requires that you specify at least 1 email address or 1 email domain

Email Addresses: l.canny3@universityofgalway.ie ×
Each email address listed here is allowed access. Adds the User.Read scope.

Email Domains: universityofgalway.ie ×
Every user whose email address ends with one of the domains listed here are allowed access. Adds the User.Read scope.

Advanced Options ▸

Scopes

Scopes: User.Read
The requested set of permissions you'd like to ask for from the identity provider. You can find the available scopes for Microsoft here.

Session

That is all!

- You should now have your
  - Authtoken
  - Edge Label (edge=abcdef123456)
  - Ngrok URL

All of which will be needed by the PYNQ board

Additional Information/Understanding

- The authtoken is used by the ngrok client on the PYNQ board to authenticate the user (i.e. login to your ngrok account on the PYNQ)

- The edge label is used to tell ngrok what endpoint will be used by the ngrok client, the endpoint means the URL that will become active and available online.

- Finally, the ngrok URL is what you will use in your browser to access the pynq board remotely (more specifically the JNB web interface)