Harder, better, faster, stronger authentication with OpenID Connect

Overview of the next-gen authentication stack for Matrix

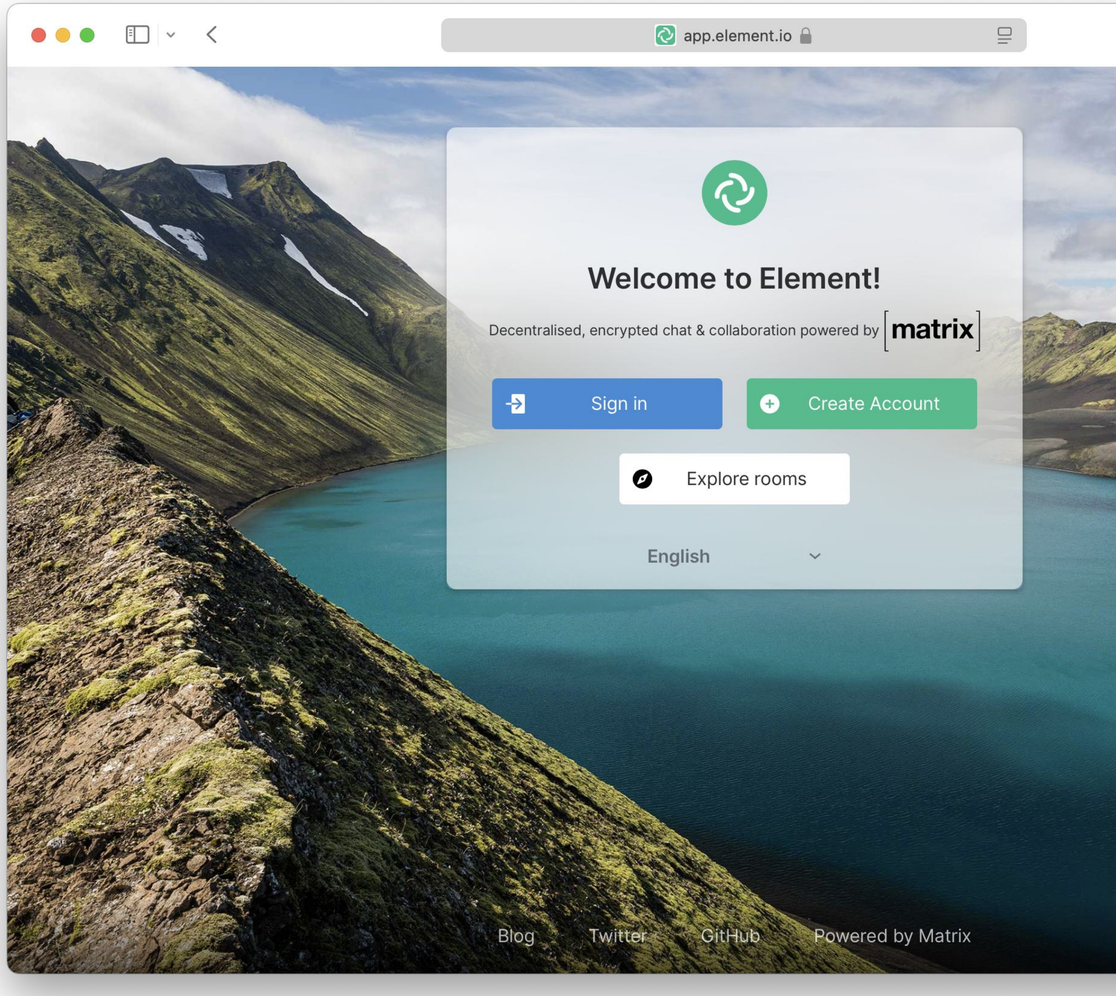
Quentin Gliech

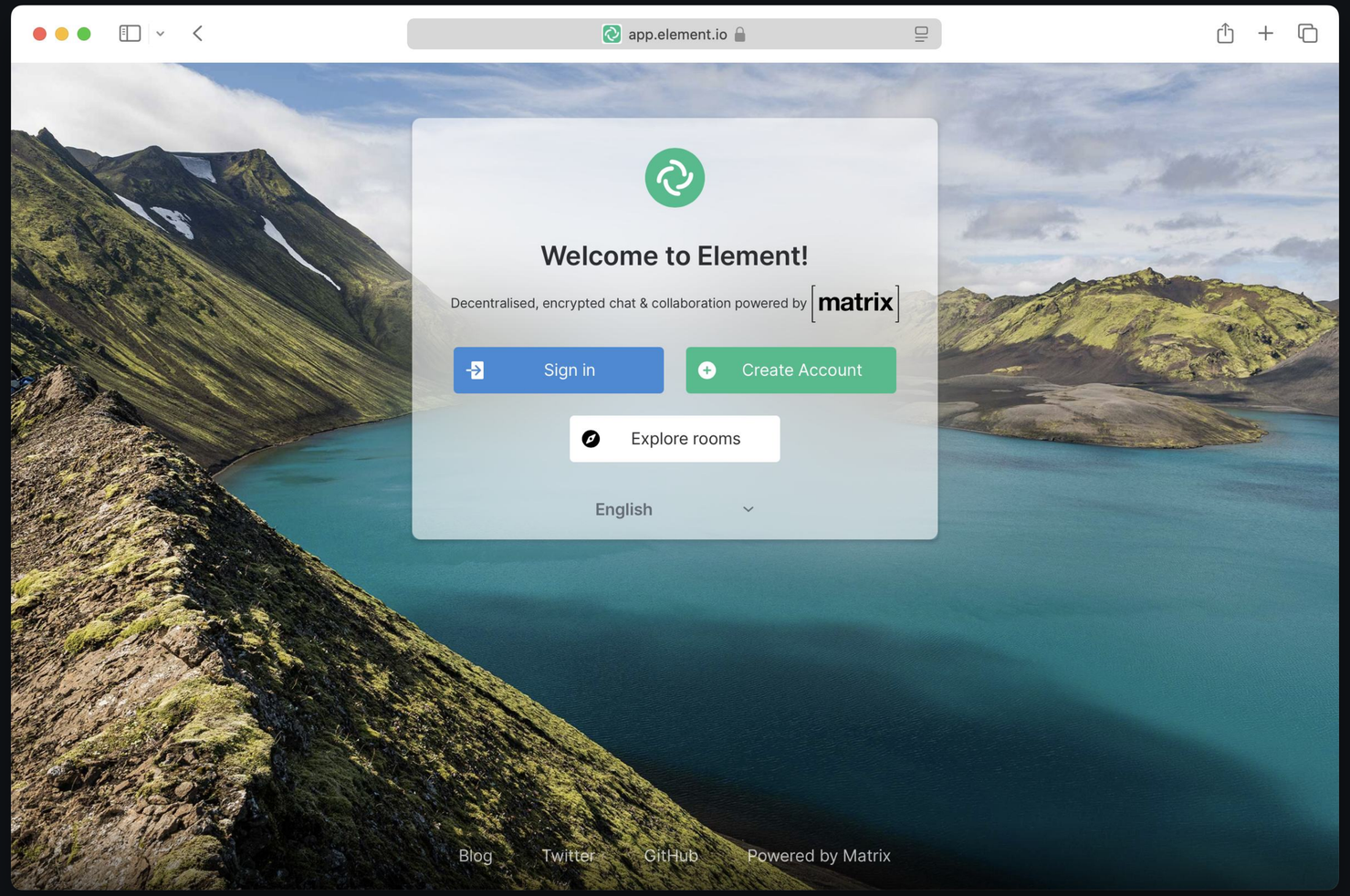
Software engineer at Element

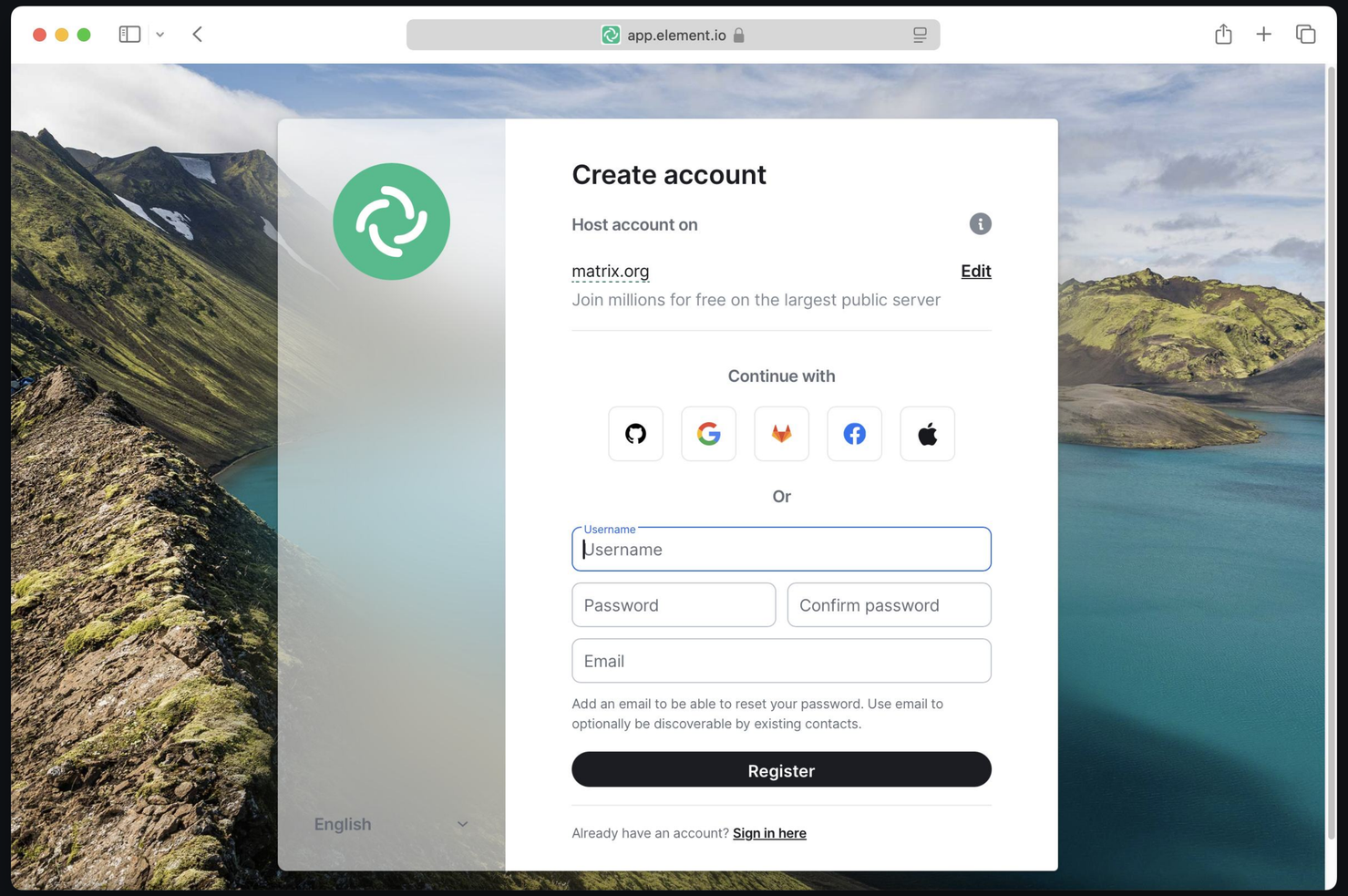
@quenting:element.io

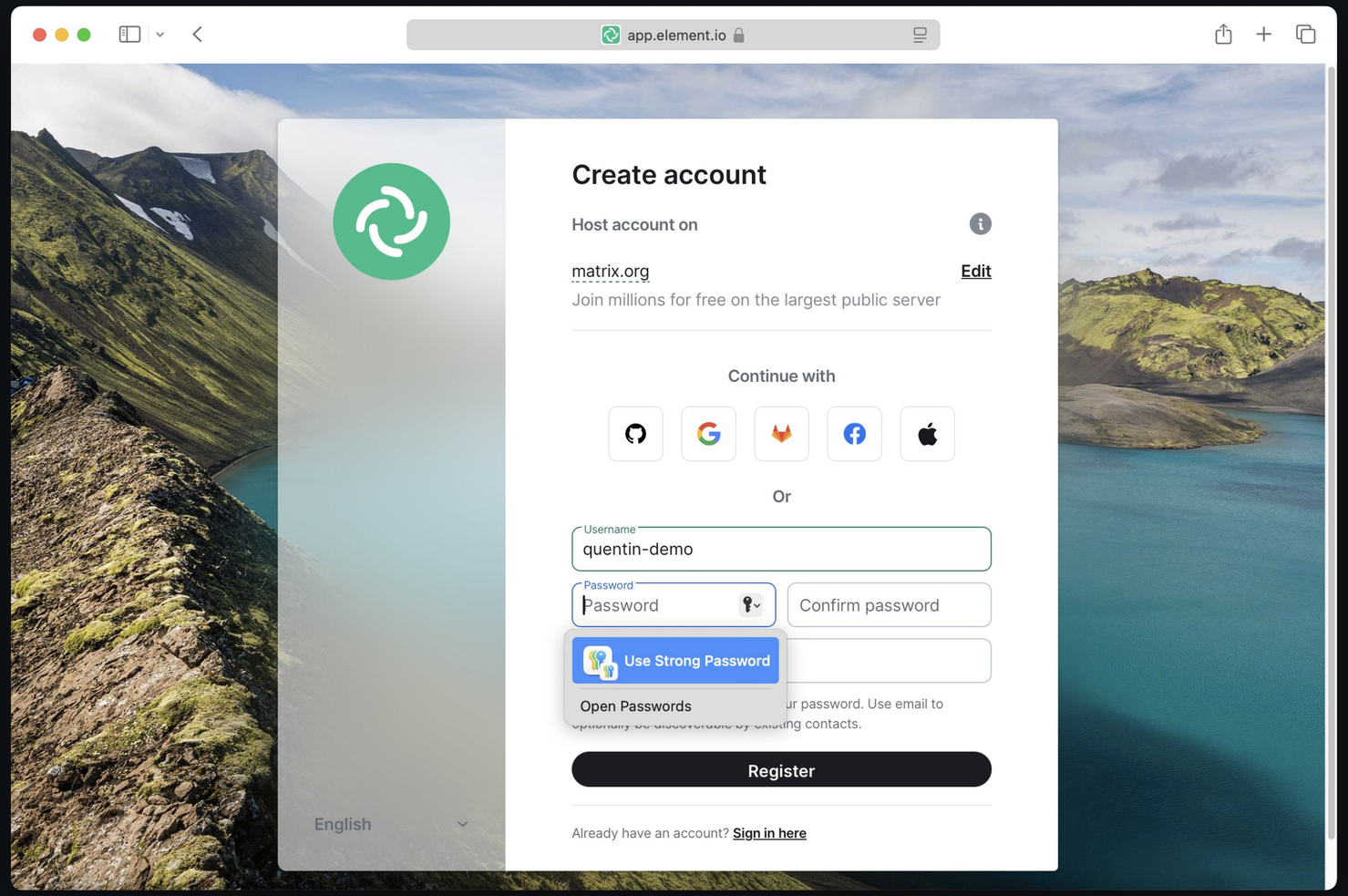
Let me show you first

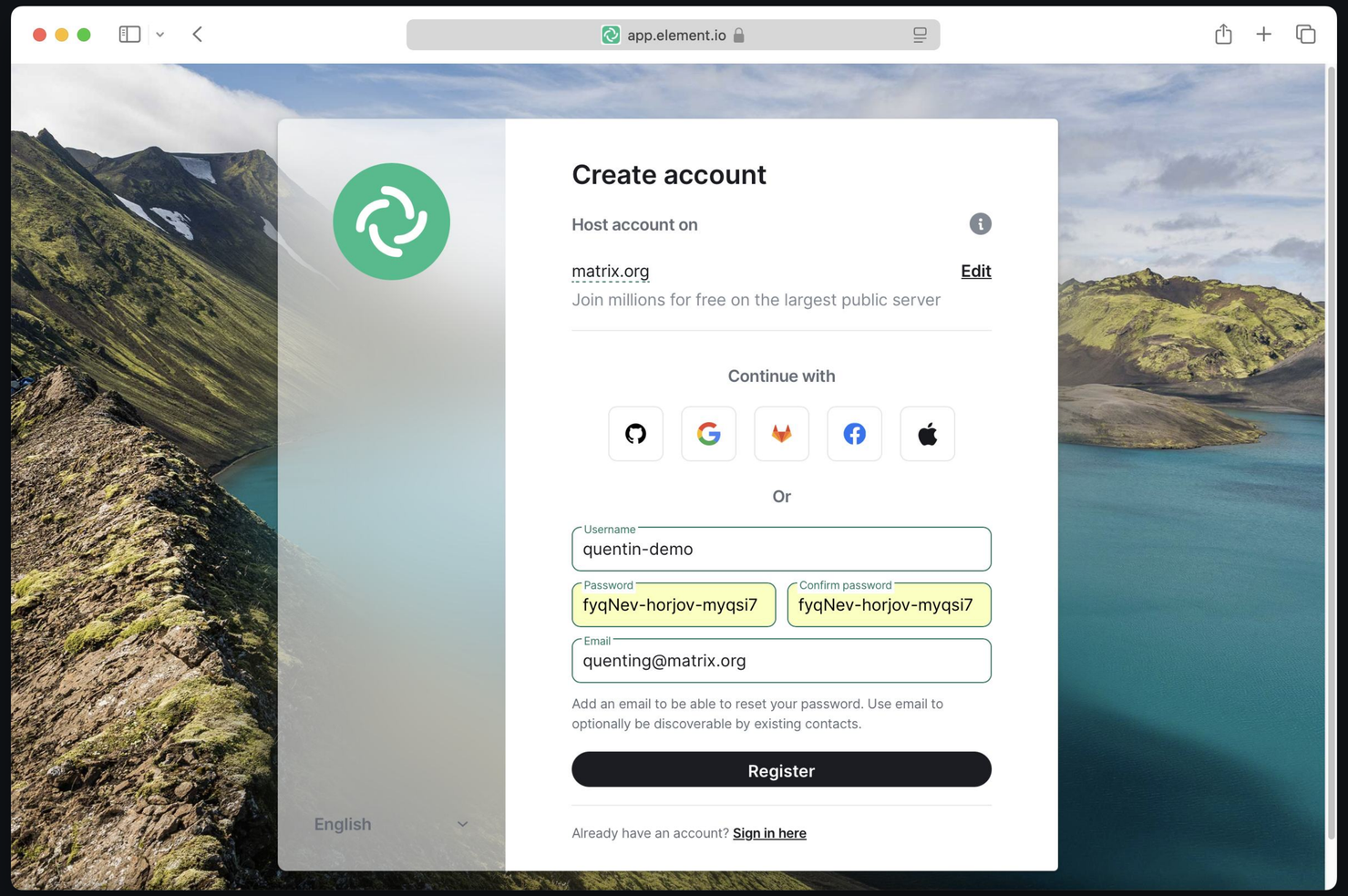
I’m a new Matrix user,
registering a new account in Element Web

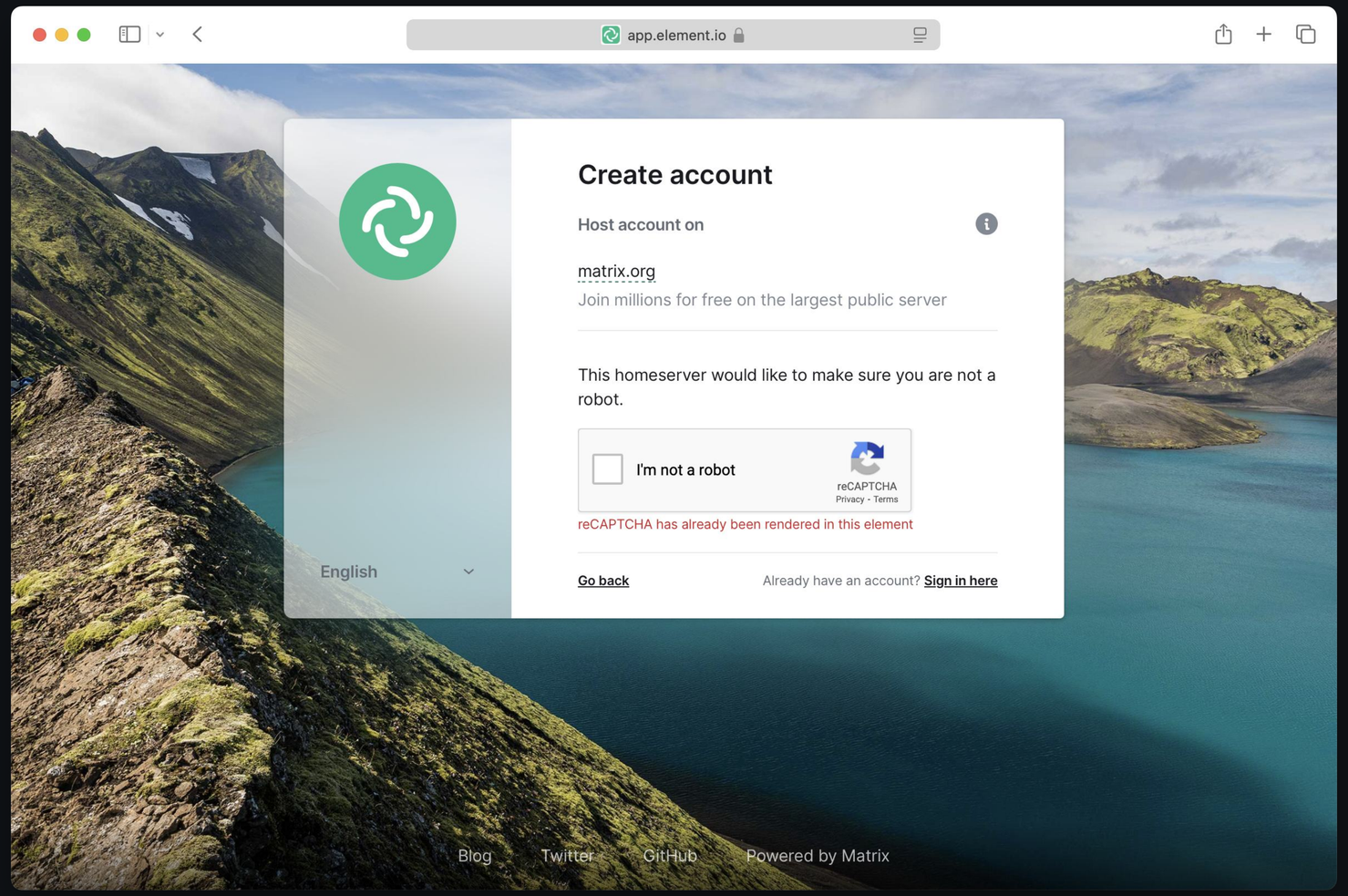


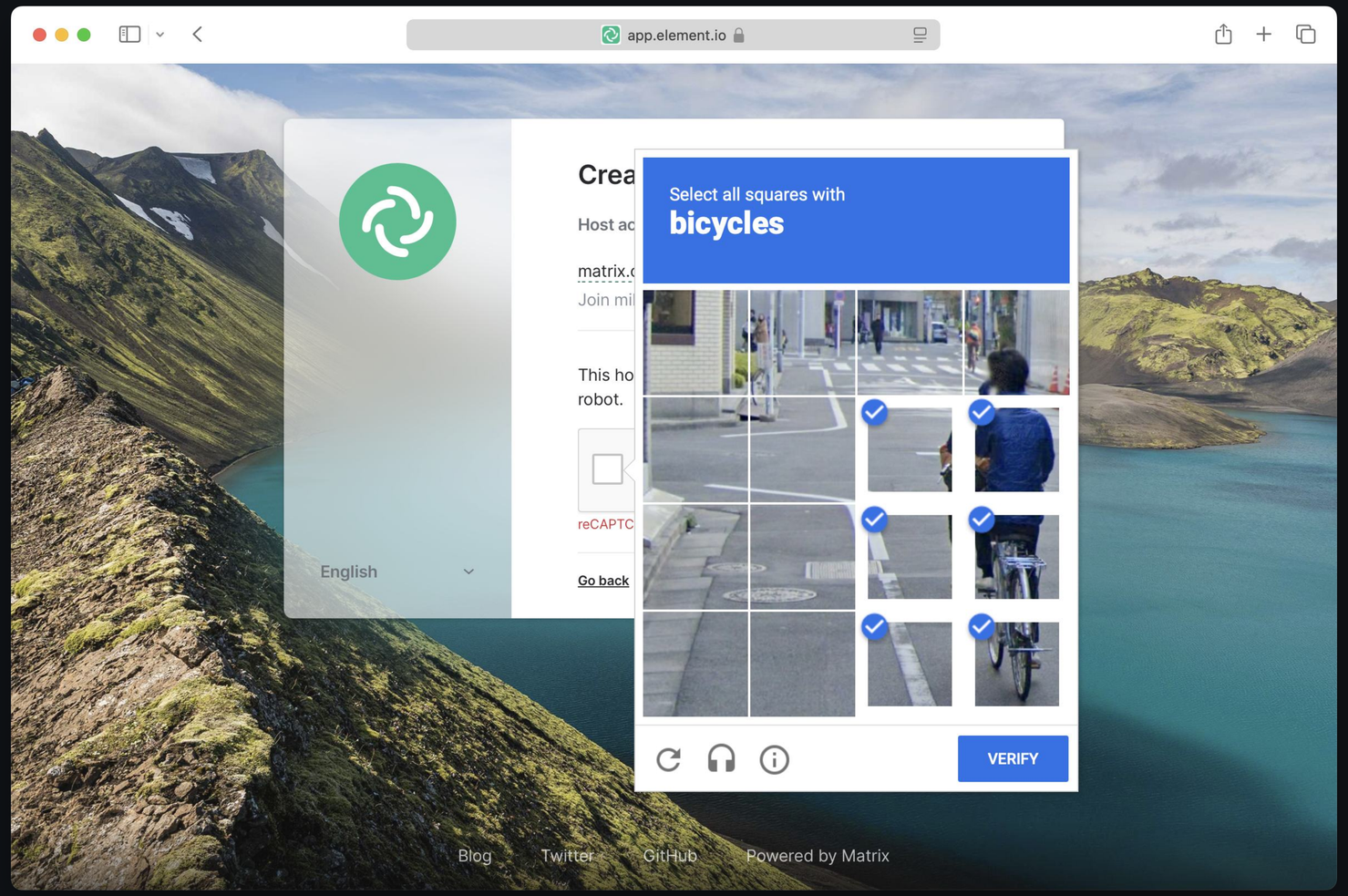

app.element.io
Welcome to Element!
Decentralised, encrypted chat & collaboration powered by matrix
Sign in
Create Account
Explore rooms
English
Blog
Twitter
GitHub
Powered by Matrix

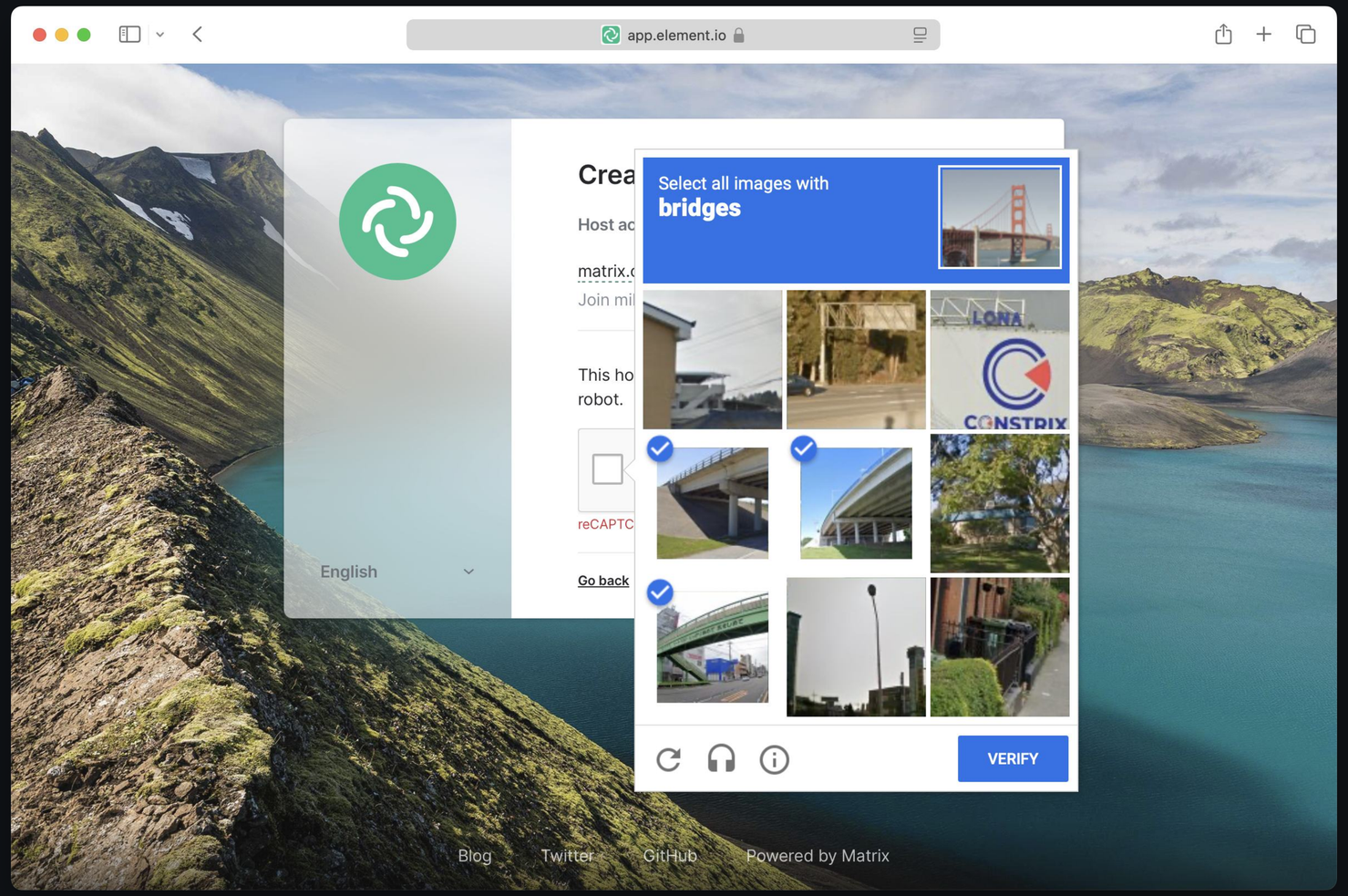

app.element.io
Create account
Host account on
matrix.org
Edit
Join millions for free on the largest public server
Continue with
Or
Username
Username
Password
Confirm password
Email
Add an email to be able to reset your password. Use email to optionally be discoverable by existing contacts.
Register
Already have an account? Sign in here
English

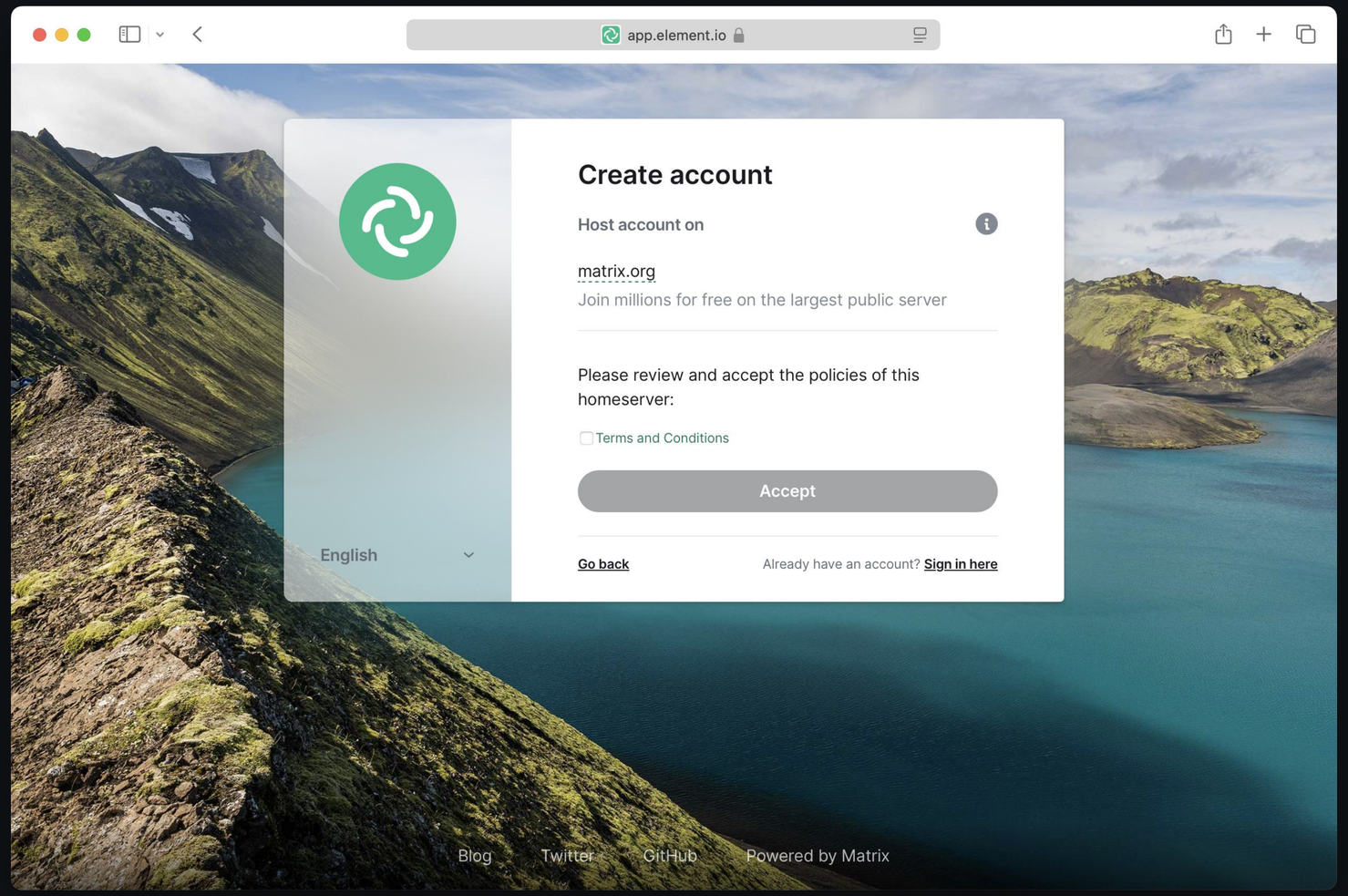

app.element.io
Create account
Host account on
matrix.org
Edit
Join millions for free on the largest public server
Continue with
Or
Username
quentin-demo
Password
Password
Confirm password
Use Strong Password
Open Passwords
Register
English
Already have an account? Sign in here

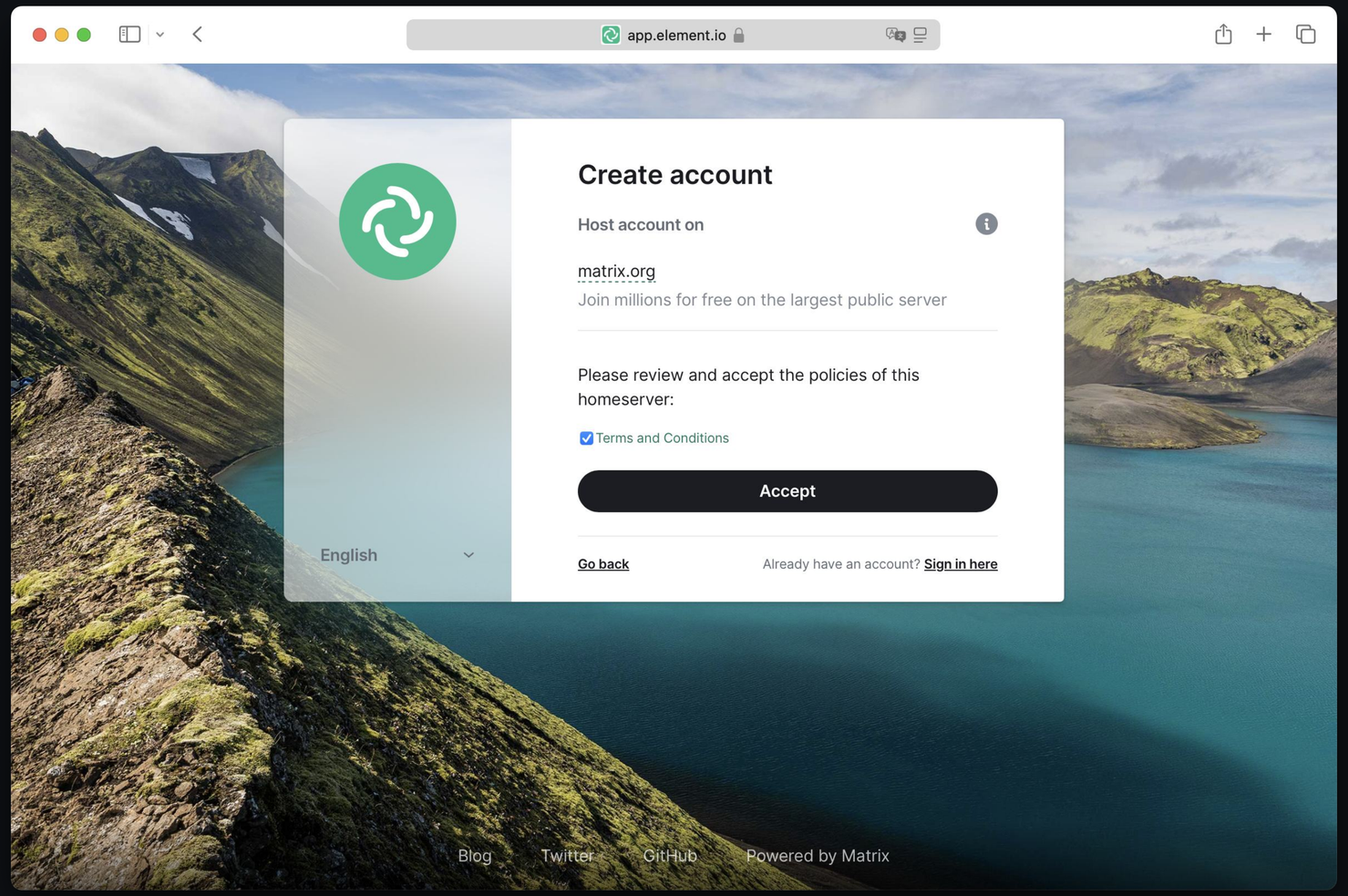

app.element.io
Create account
Host account on
matrix.org
Edit
Join millions for free on the largest public server
Continue with
Or
Username
quentin-demo
Password
fyqNev-horjov-myqsi7
Confirm password
fyqNev-horjov-myqsi7
Email
quenting@matrix.org
Add an email to be able to reset your password. Use email to optionally be discoverable by existing contacts.
Register
English
Already have an account? Sign in here

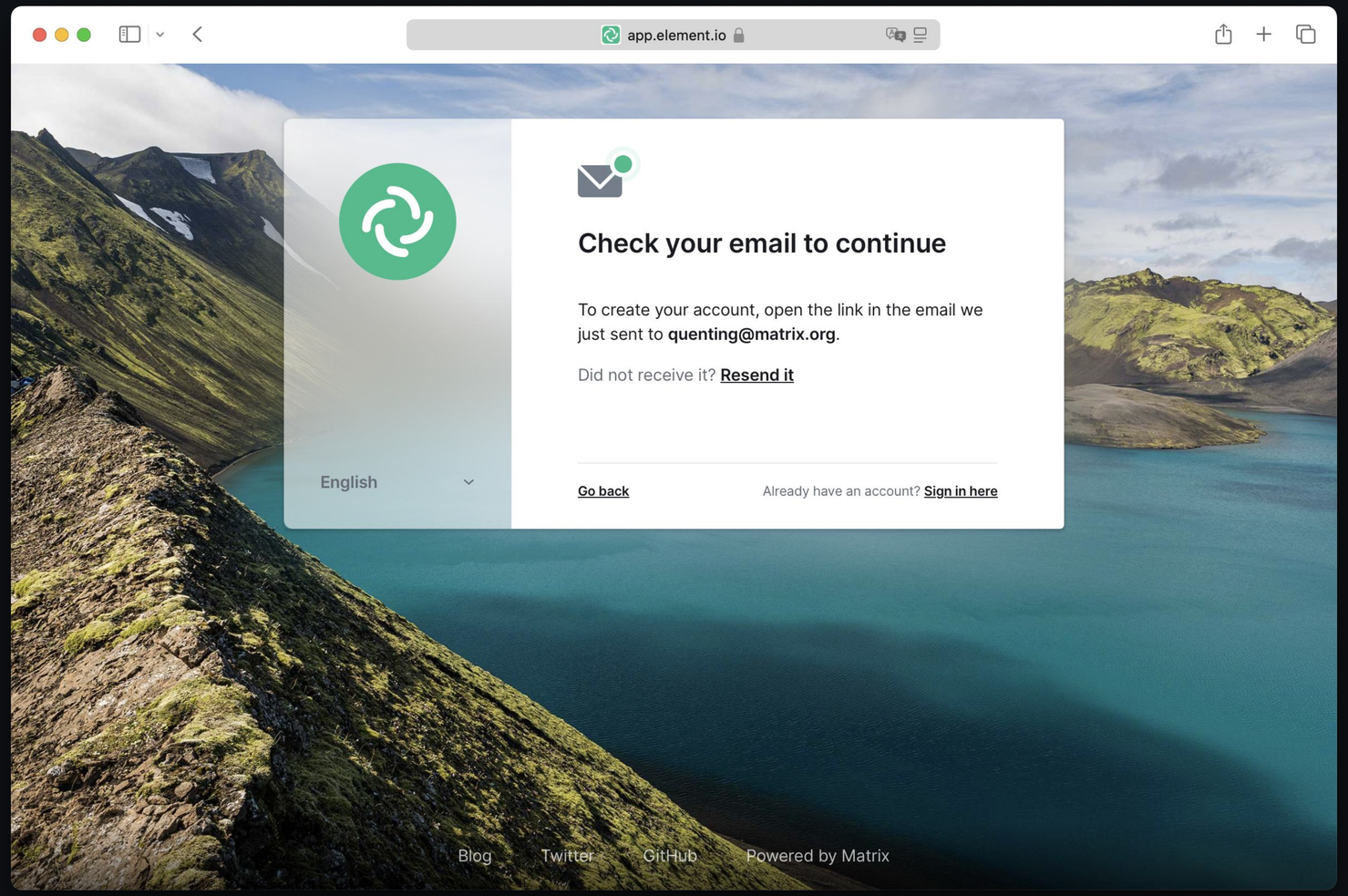

app.element.io
Create account
Host account on
matrix.org
Join millions for free on the largest public server
This homeserver would like to make sure you are not a robot.
I'm not a robot
reCAPTCHA
Privacy - Terms
reCAPTCHA has already been rendered in this element
English
Go back
Already have an account? Sign in here
Blog
Twitter
GitHub
Powered by Matrix

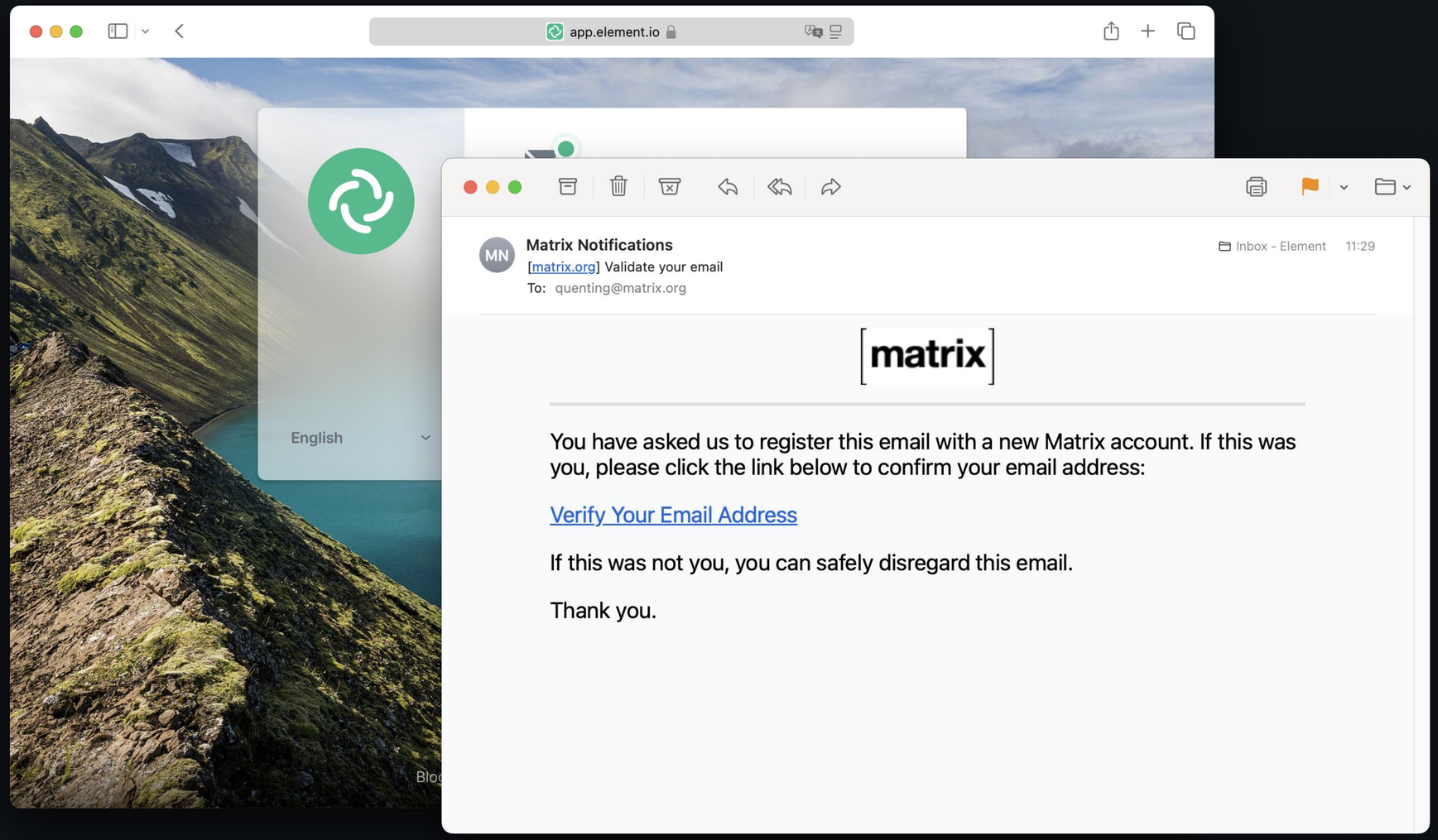

app.element.io
Crea
Host ac
matrix.
Join mi
This ho
robot.
reCAPTC
English
Go back
Select all squares with
bicycles
VERIFY
Blog
Twitter
GitHub
Powered by Matrix

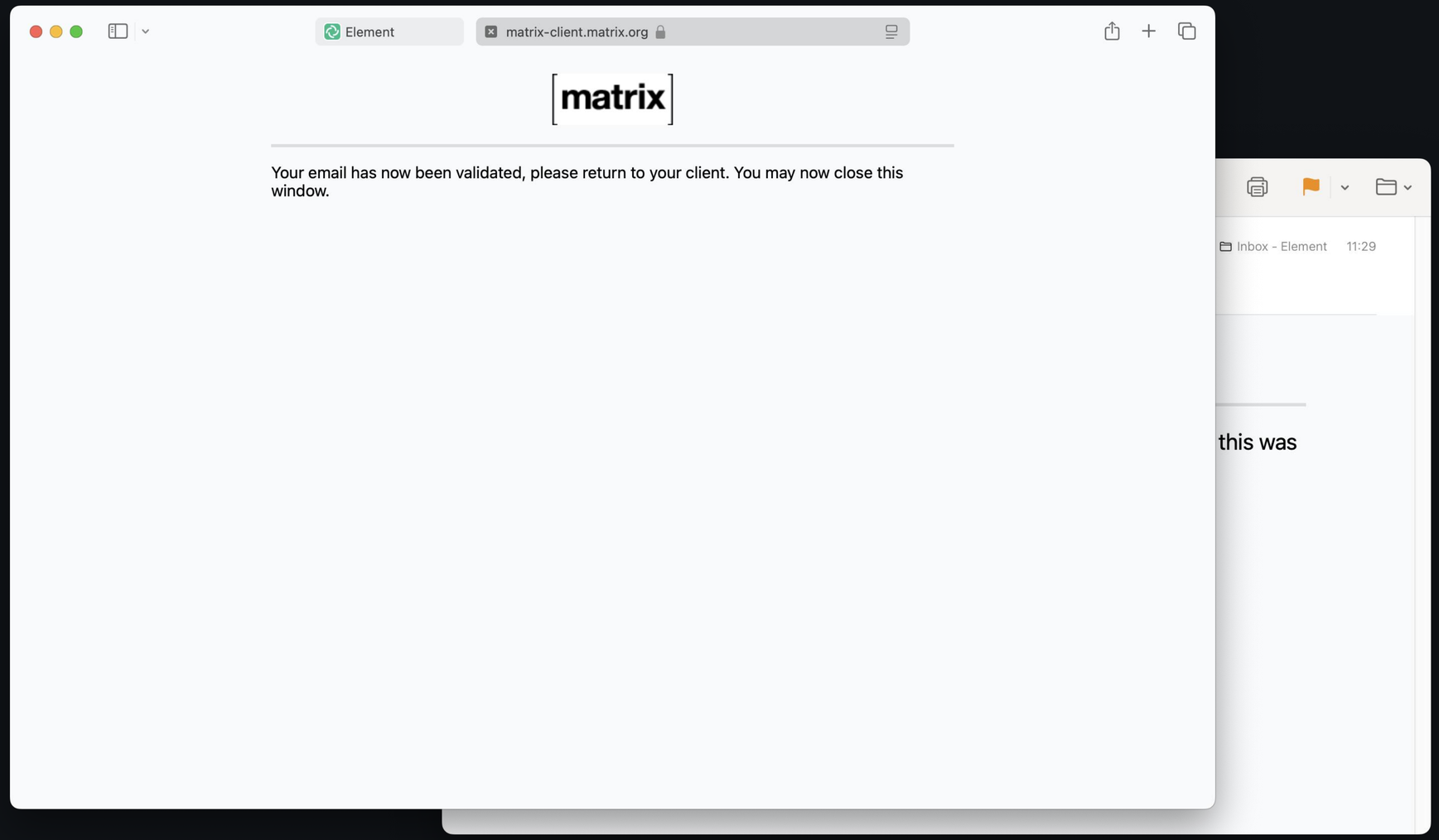

app.element.io
Crea
Host ac
matrix.
Join mi
This ho
robot.
reCAPTC
English
Go back
Select all images with
bridges
LONA
CONSTRIX
VERIFY
Blog
Twitter
GitHub
Powered by Matrix

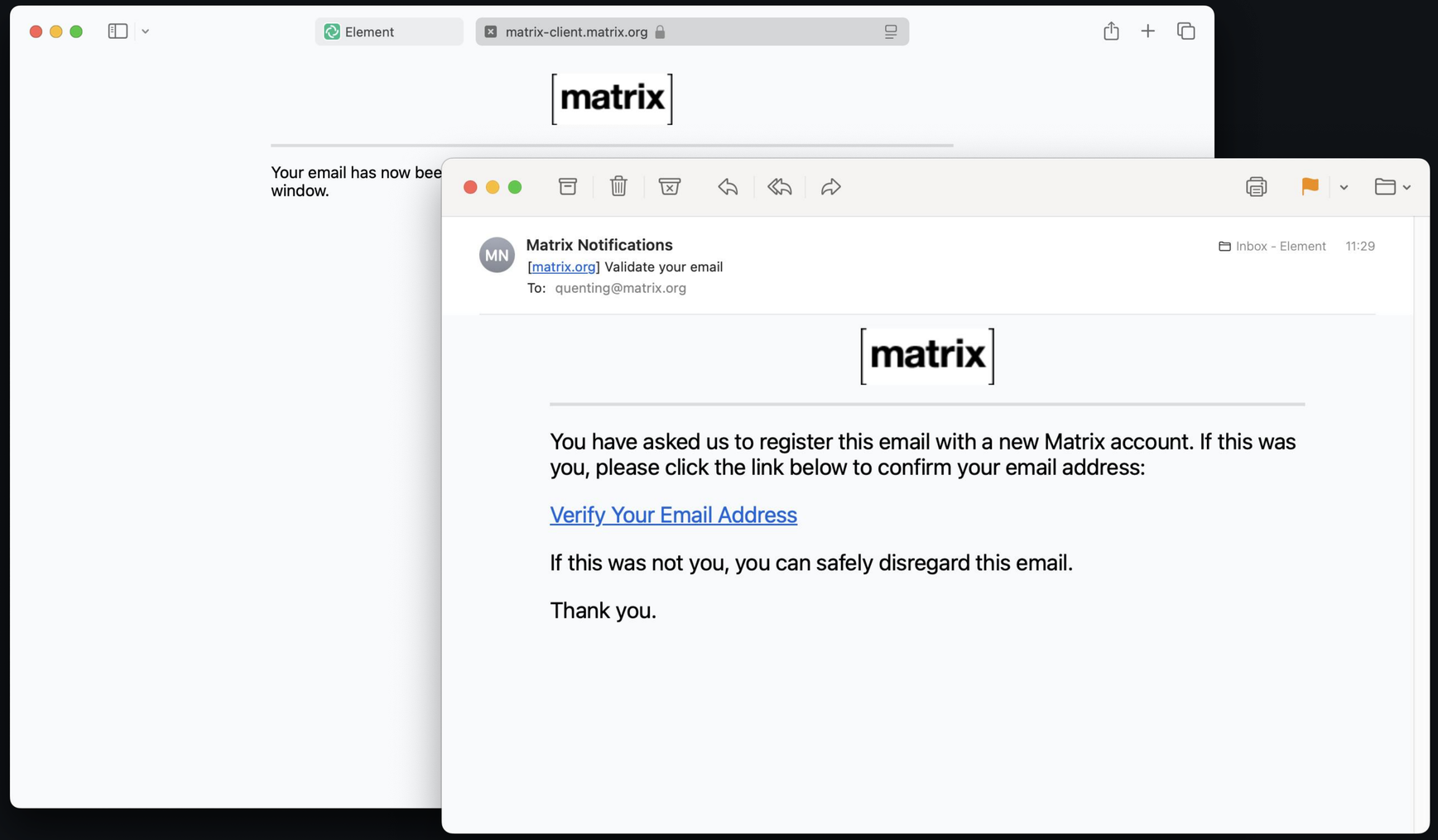

app.element.io
Create account
Host account on
matrix.org
Join millions for free on the largest public server
Please review and accept the policies of this homeserver:
Terms and Conditions
Accept
English
Go back
Already have an account? Sign in here
Blog
Twitter
GitHub
Powered by Matrix

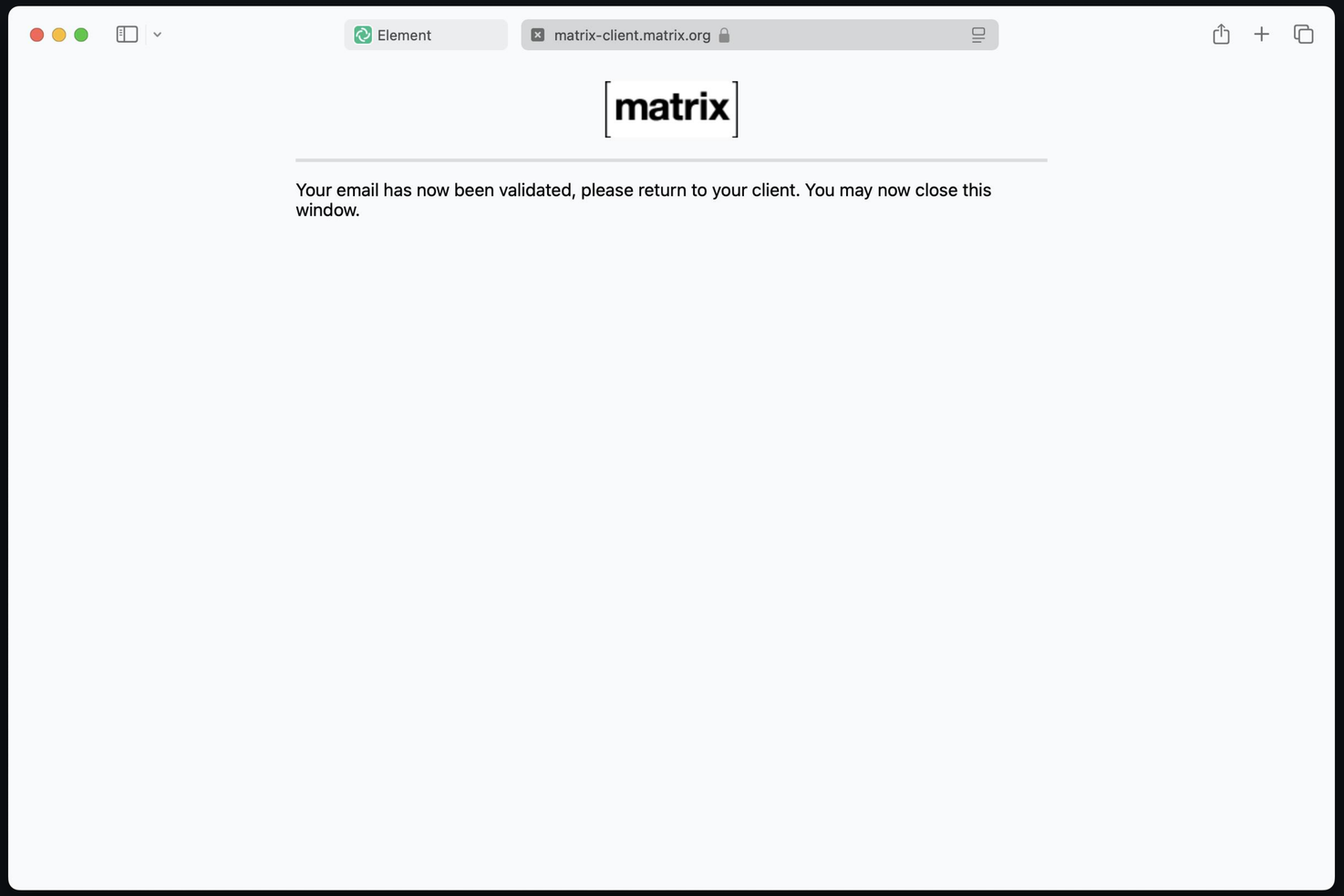

app.element.io
Create account
Host account on
matrix.org
Join millions for free on the largest public server
Please review and accept the policies of this homeserver:
Terms and Conditions
Accept
English
Go back
Already have an account? Sign in here
Blog
Twitter
GitHub
Powered by Matrix


app.element.io
Check your email to continue
To create your account, open the link in the email we just sent to quenting@matrix.org.
Did not receive it? Resend it
English
Go back
Already have an account? Sign in here
Blog
Twitter
GitHub
Powered by Matrix


app.element.io
English
MN
Matrix Notifications
[matrix.org] Validate your email
To: quenting@matrix.org
Inbox - Element
11:29
matrix
You have asked us to register this email with a new Matrix account. If this was you, please click the link below to confirm your email address:
Verify Your Email Address
If this was not you, you can safely disregard this email.
Thank you.


Element
matrix-client.matrix.org
matrix
Your email has now been validated, please return to your client. You may now close this window.
Inbox - Element
11:29
this was


Element
matrix-client.matrix.org
matrix
Your email has now bee
window.
MN
Matrix Notifications
[matrix.org] Validate your email
To: quenting@matrix.org
Inbox - Element
11:29
matrix
You have asked us to register this email with a new Matrix account. If this was you, please click the link below to confirm your email address:
Verify Your Email Address
If this was not you, you can safely disregard this email.
Thank you.


Element
matrix-client.matrix.org
matrix
Your email has now been validated, please return to your client. You may now close this window.

app.element.io

Search ⌘ K

Home

Welcome

People

Rooms

Welcome to Element.

With free end-to-end encrypted messaging, and unlimited voice and video calls, Element is a great way to stay in touch.

Start your first chat

Only 4 steps to go

Complete these to get the most out of Element

Create account
You made it!

2 Find and invite your friends
It's what you're here for, so lets get to it

Find friends

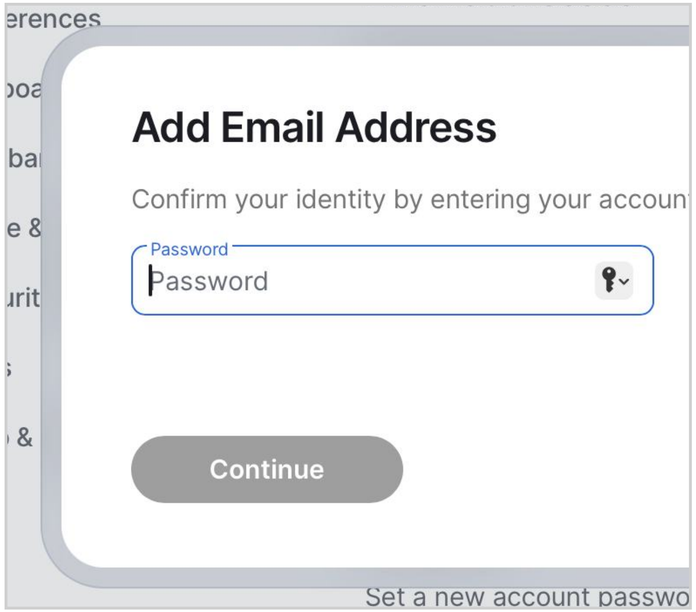
3 Download Element
Don't miss a thing by taking Element with you

Download apps

How did we end up here?

A gentle introduction to
User-Interactive Authentication

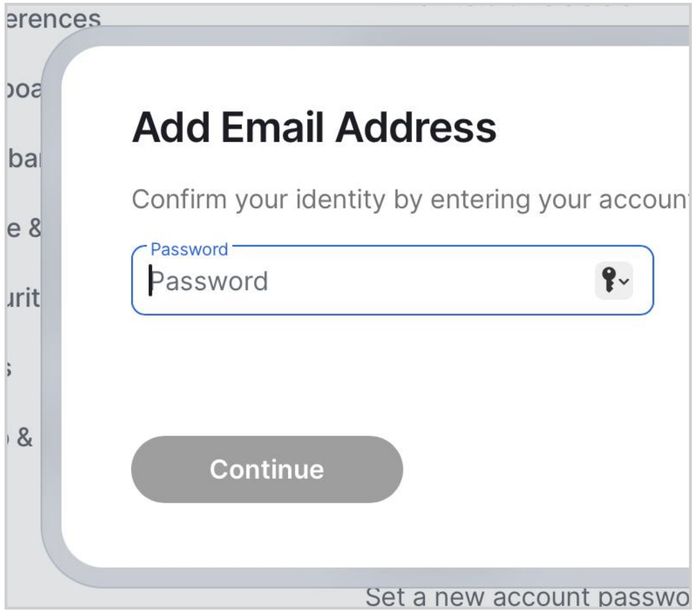
A gentle introduction to User-Interactive Authentication



Protect sensitive operations

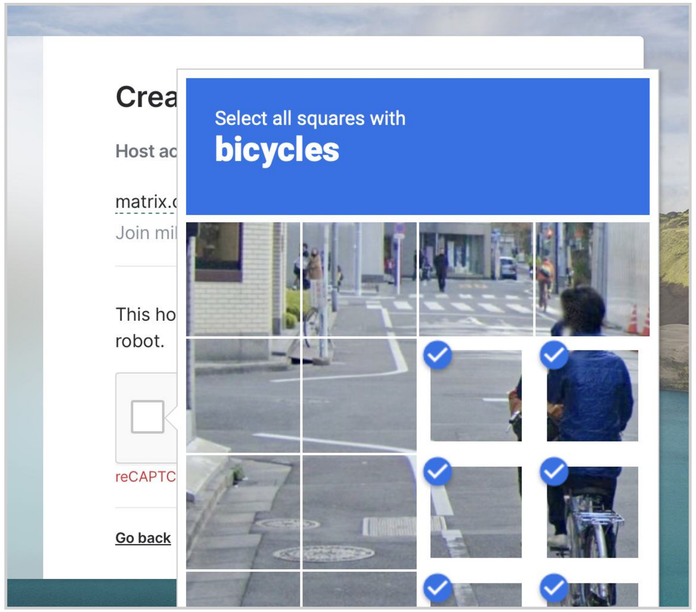
Account deactivation, password change, adding/removing an email-address, etc.

A gentle introduction to User-Interactive Authentication



Protect sensitive operations

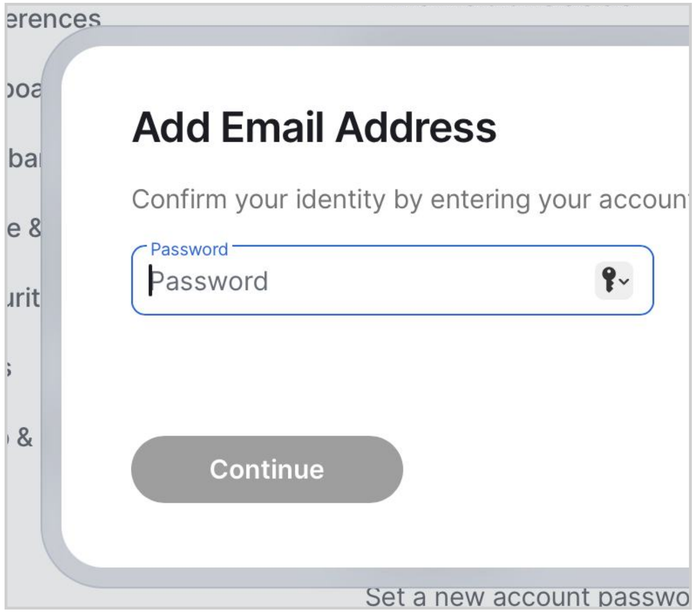
Account deactivation, password change, adding/removing an email-address, etc.



Multi-stage authentication

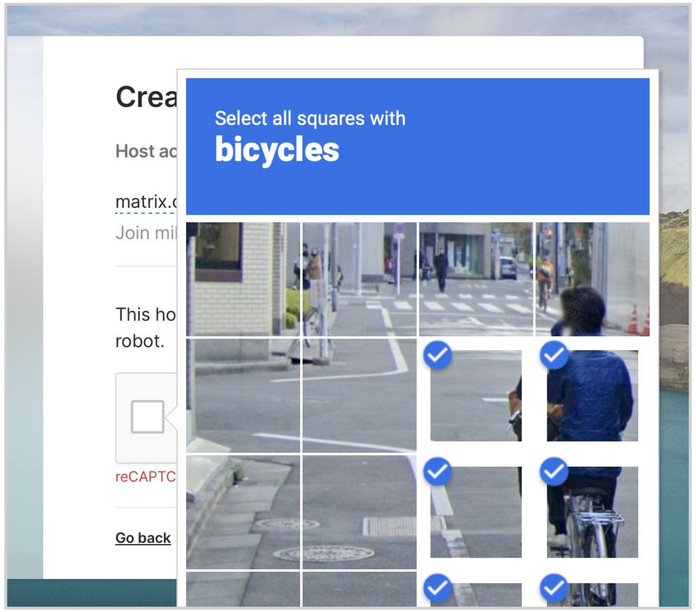
Dynamically ask for multiple steps

A gentle introduction to User-Interactive Authentication



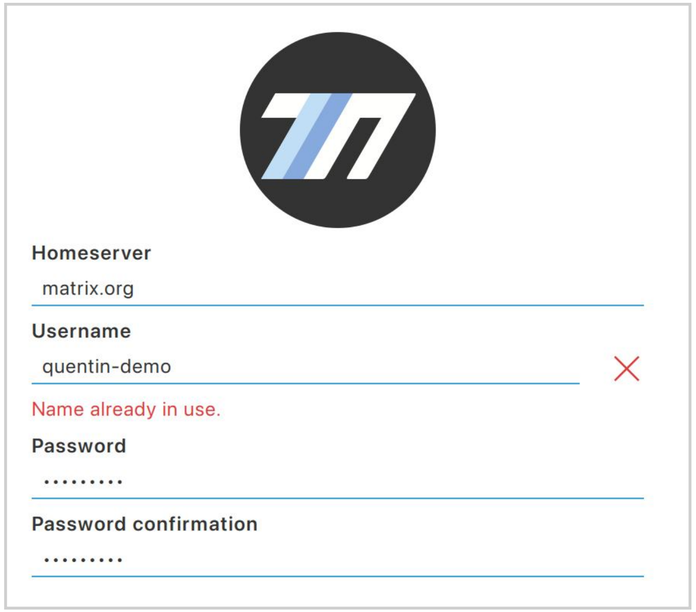
Protect sensitive operations

Account deactivation, password change, adding/removing an email-address, etc.



Multi-stage authentication

Dynamically ask for multiple steps

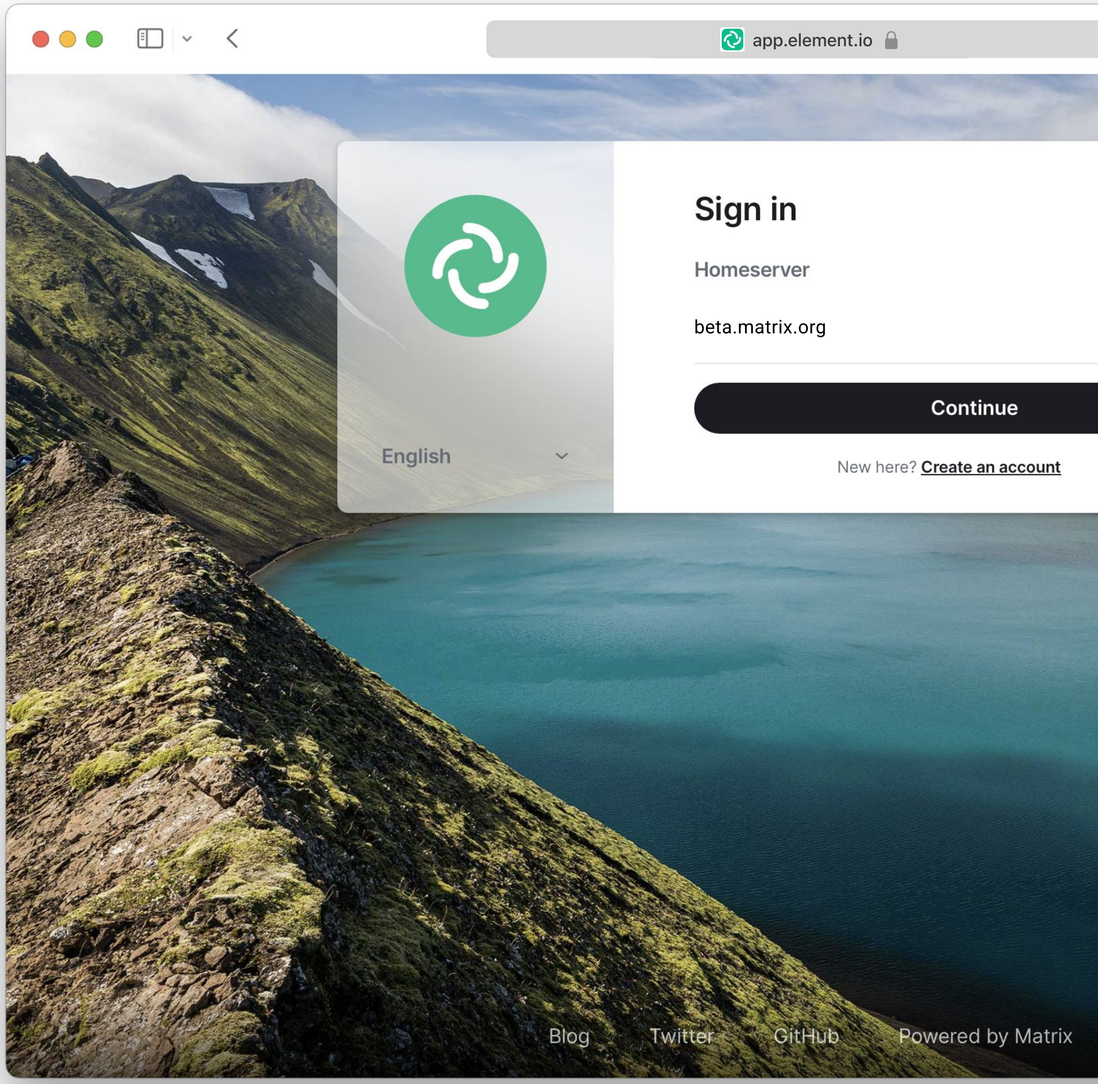


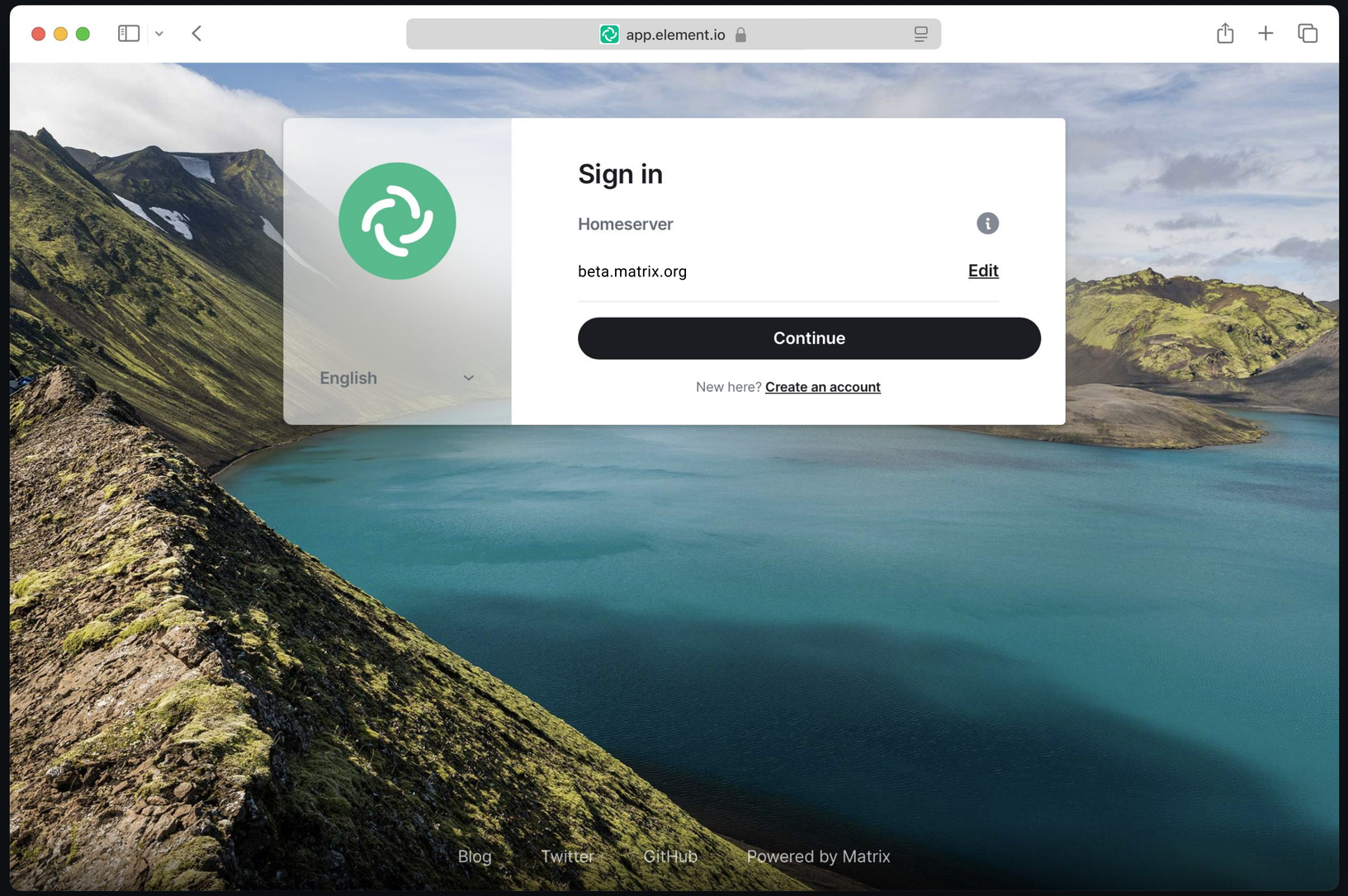
Client-native user interface

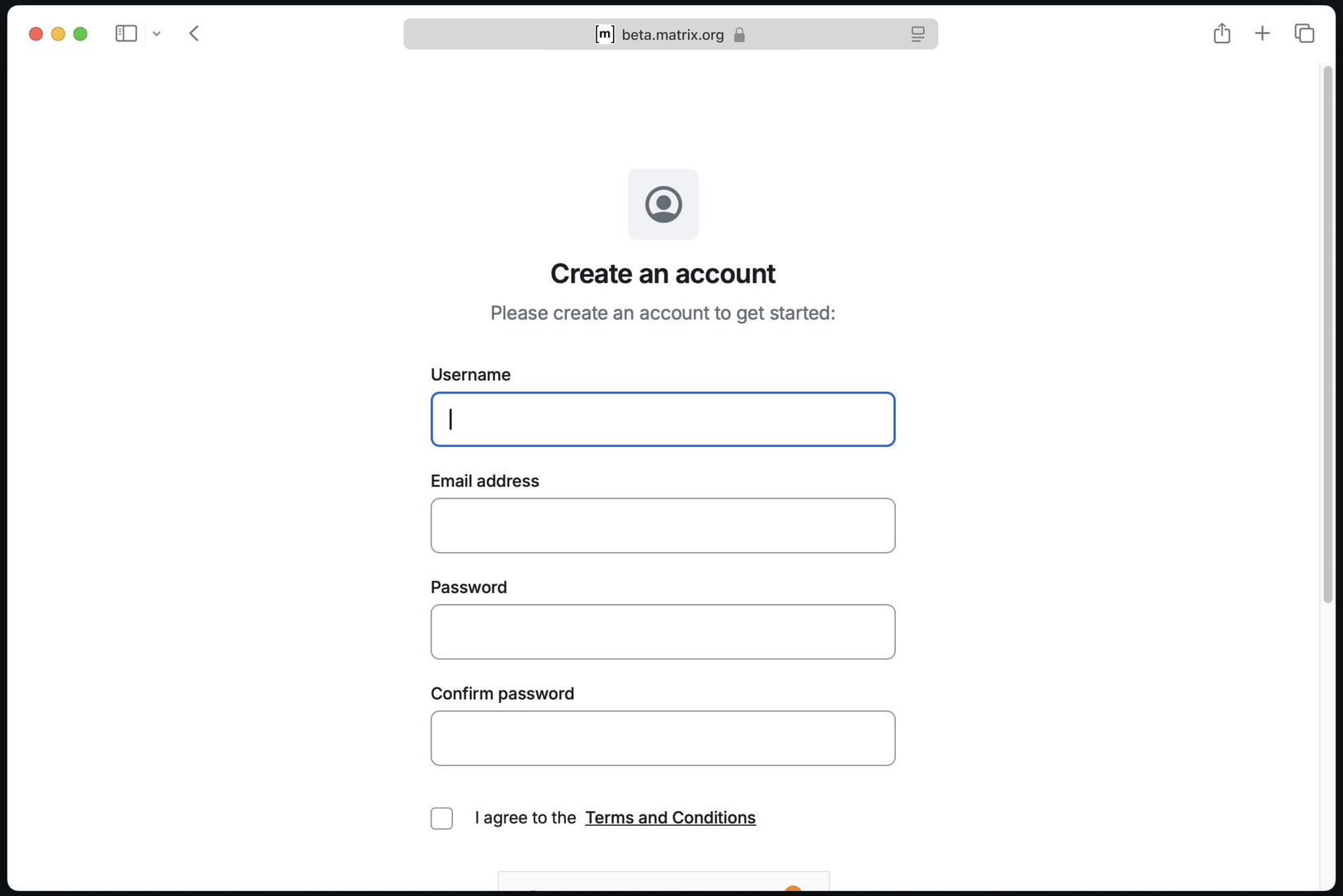
Each step is ideally implemented within the client, in a streamlined UI

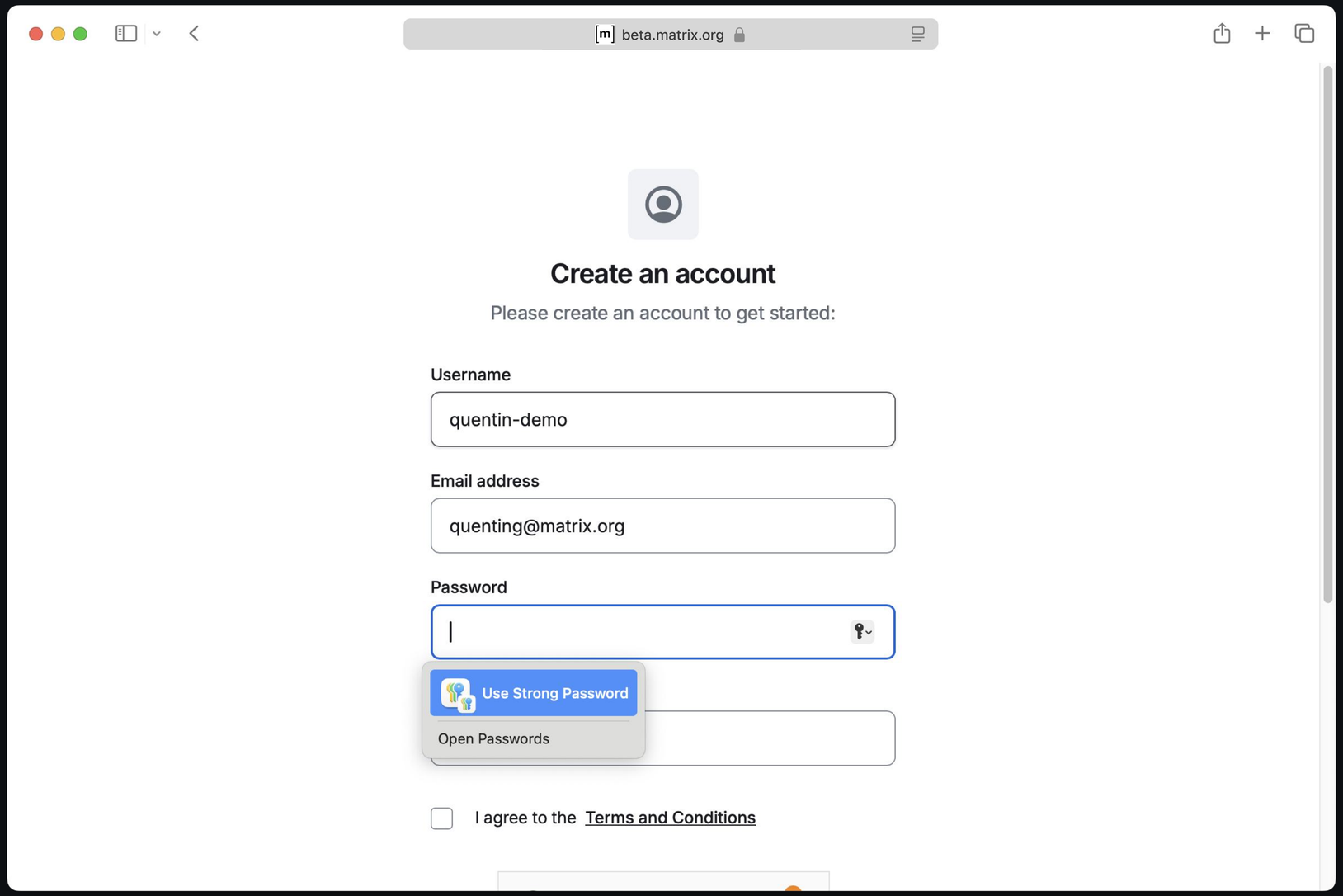
Same thing, with the new authentication stack

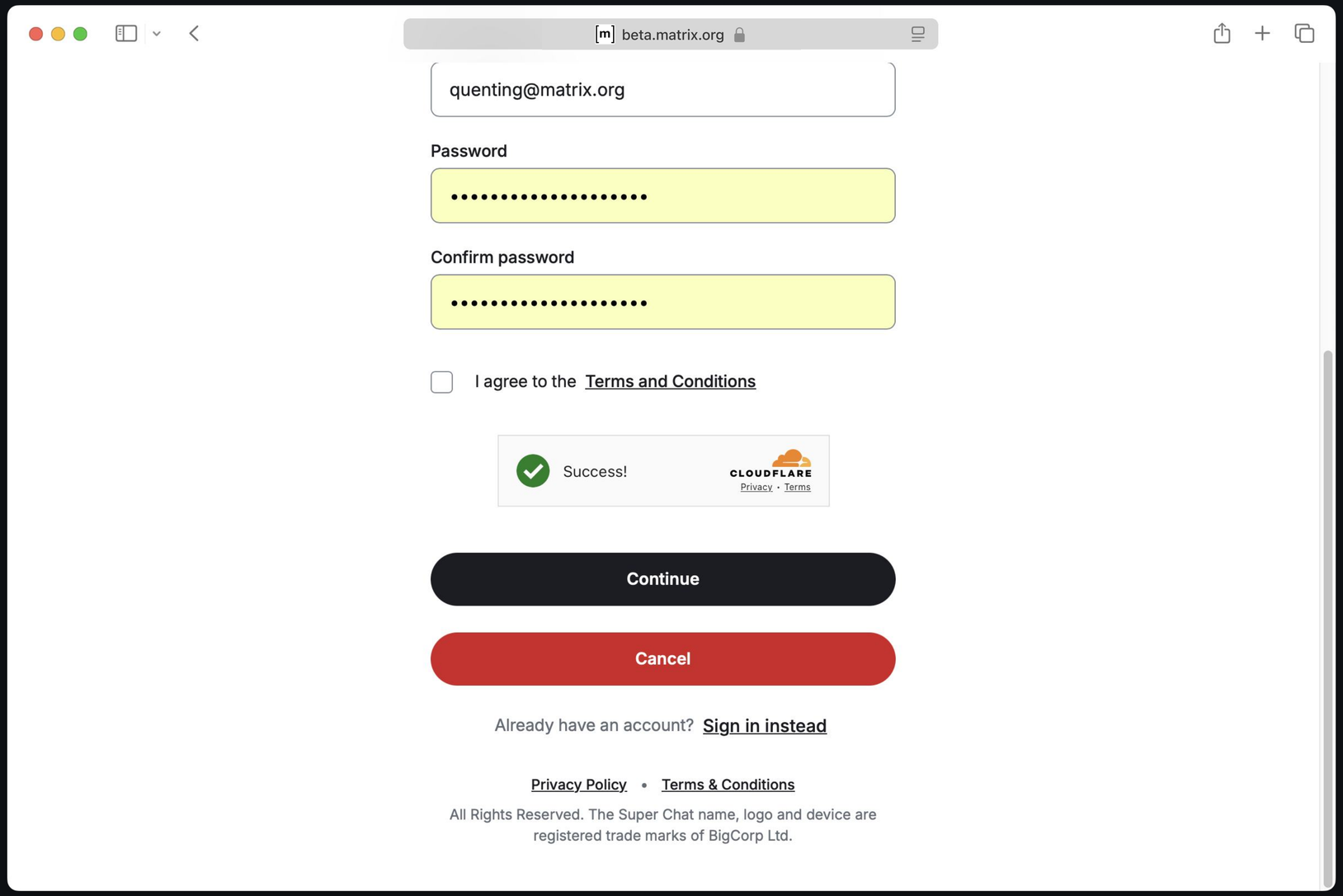
I’m a new Matrix user,
registering a new account in Element Web

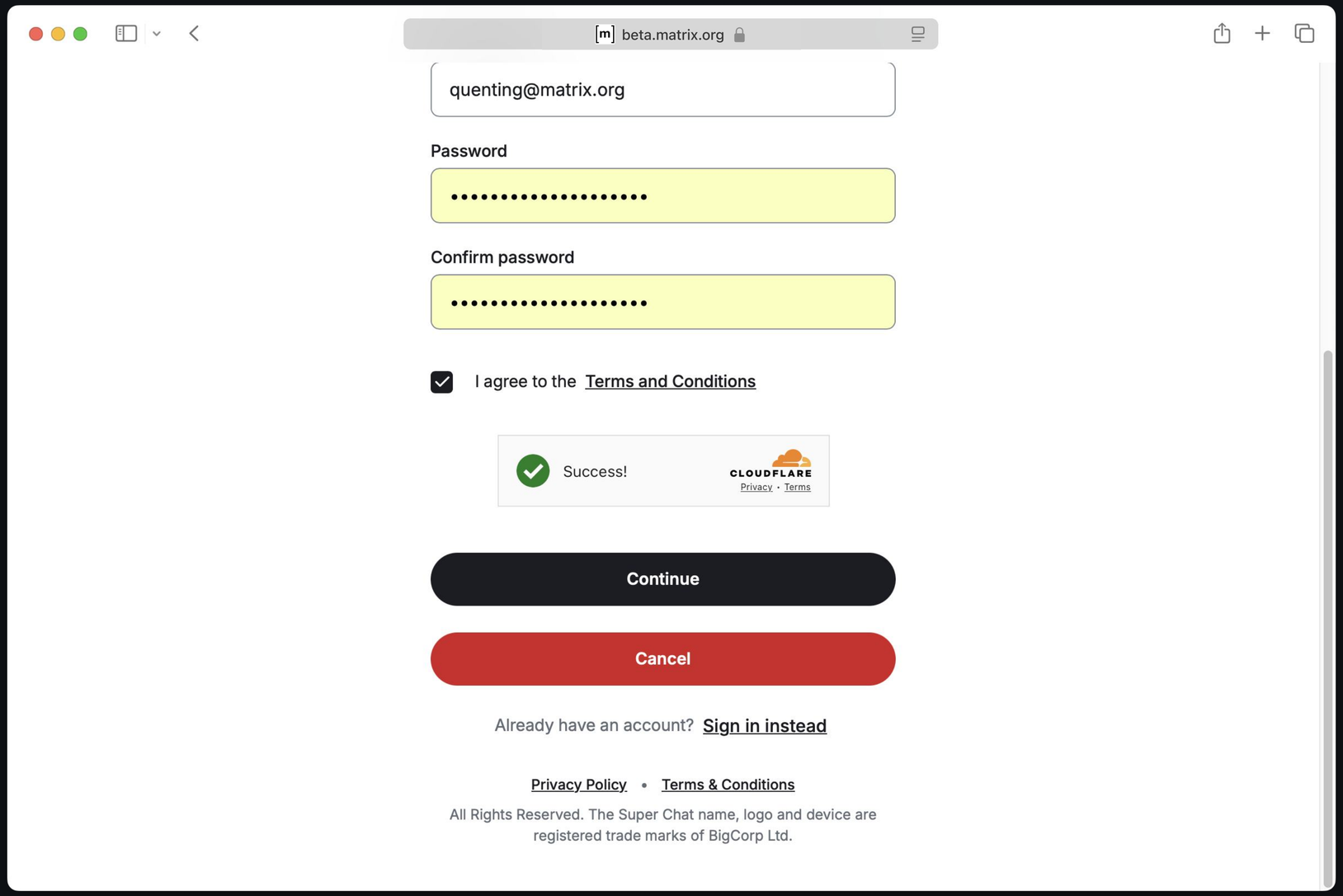


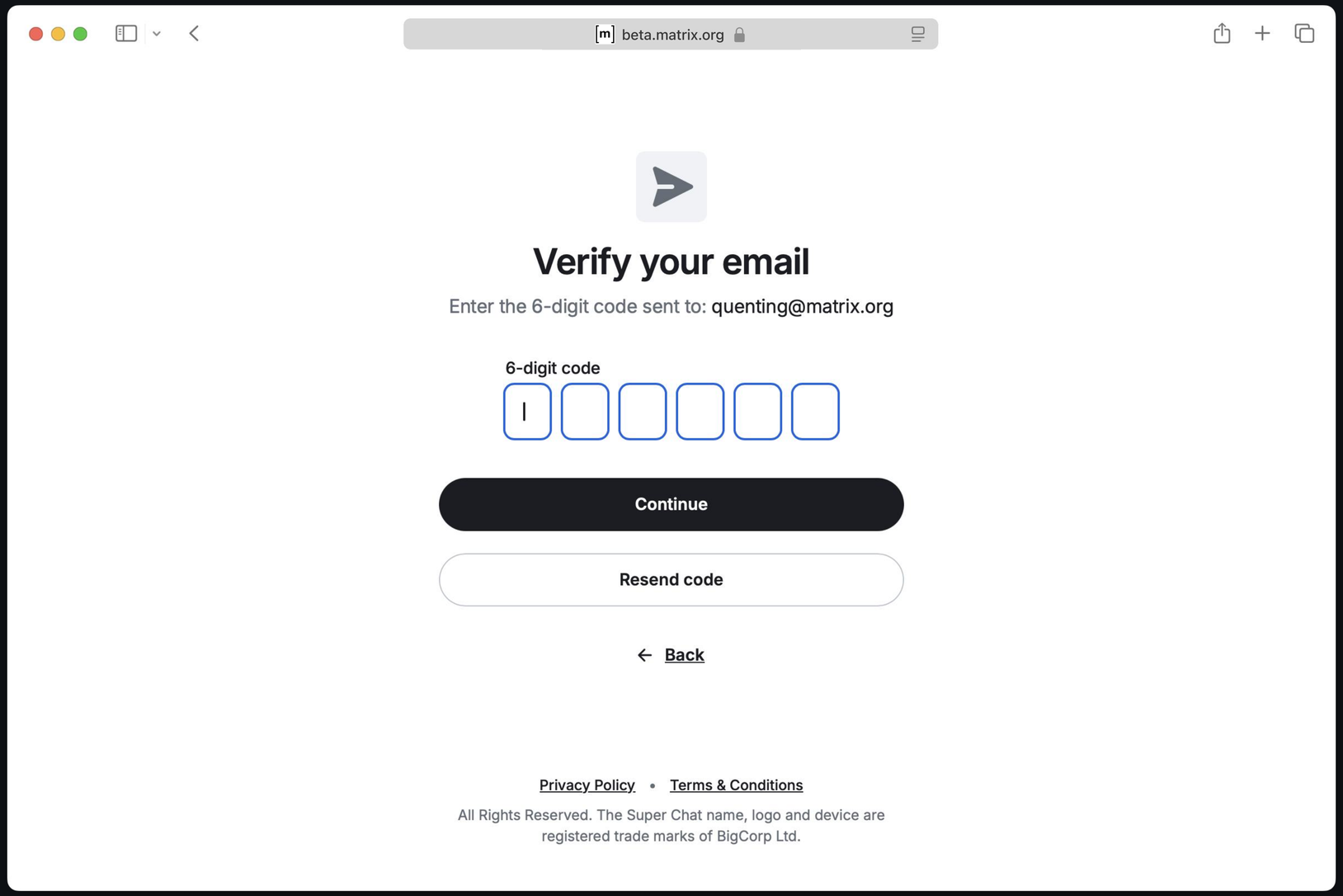

app.element.io
Sign in
Homeserver
beta.matrix.org
Edit
Continue
New here? Create an account
English
Blog
Twitter
GitHub
Powered by Matrix

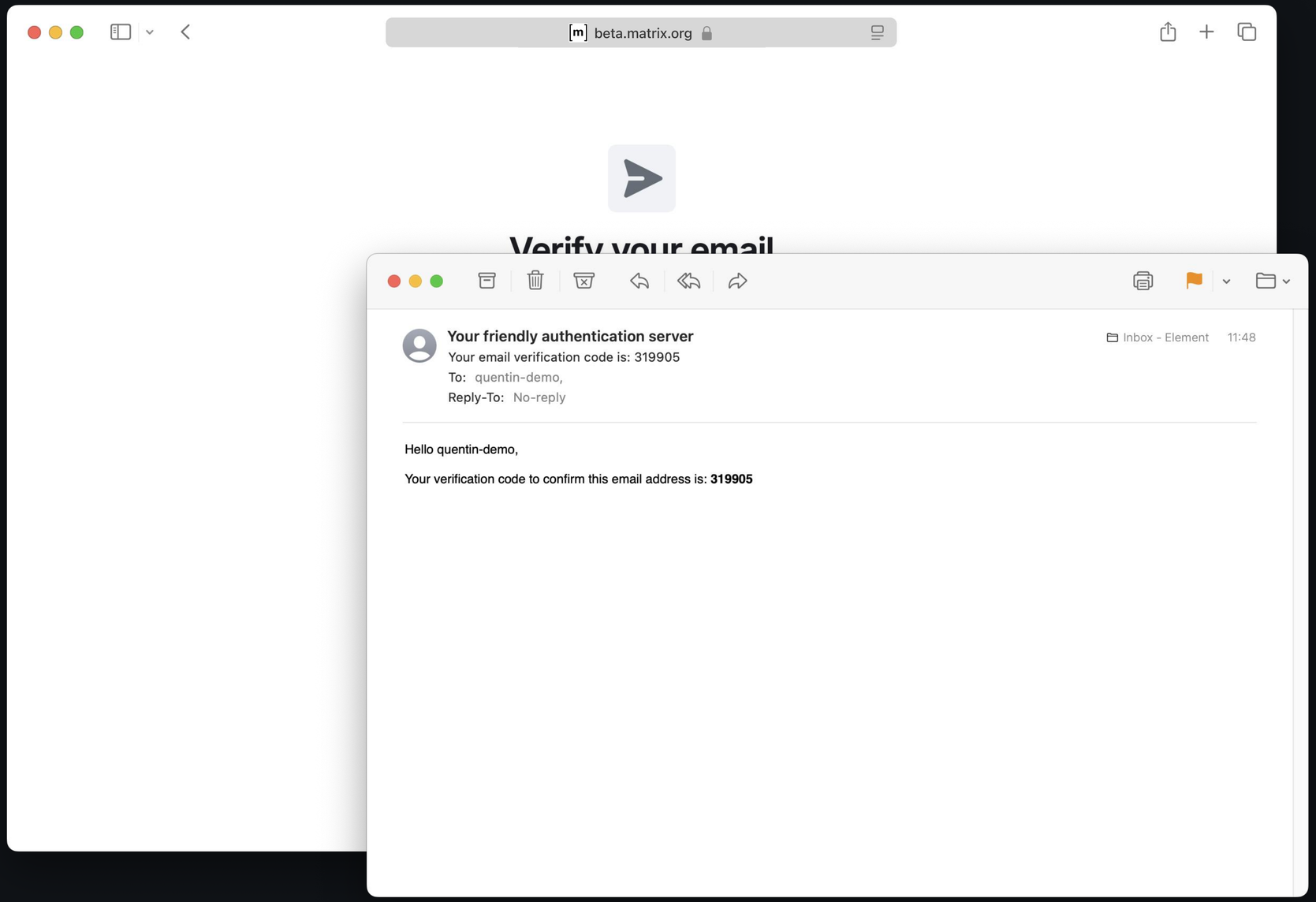

beta.matrix.org
Create an account
Please create an account to get started:
Username
Email address
Password
Confirm password
I agree to the Terms and Conditions

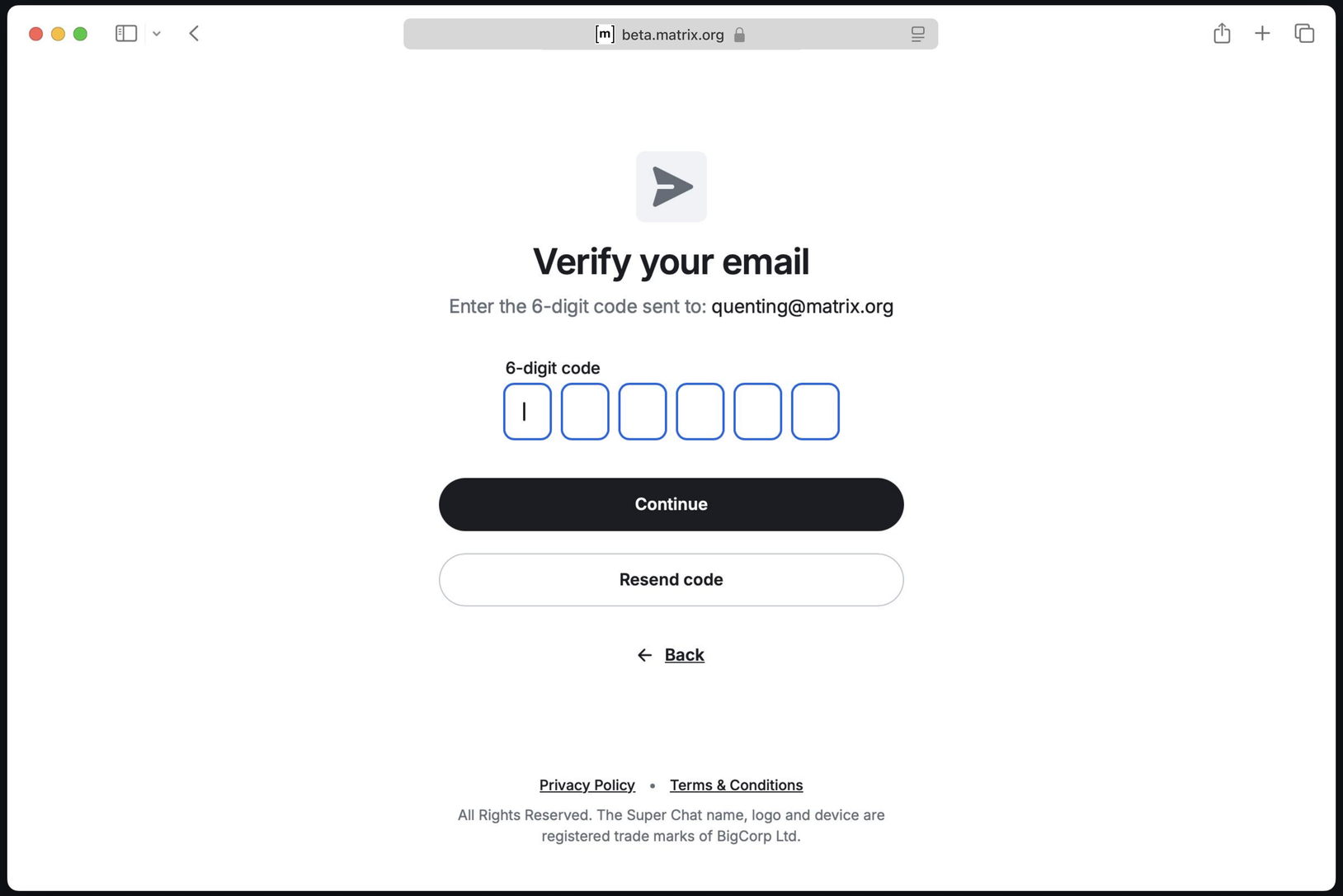

beta.matrix.org
Create an account
Please create an account to get started:
Username
quentin-demo
Email address
quenting@matrix.org
Password
Use Strong Password
Open Passwords
I agree to the Terms and Conditions

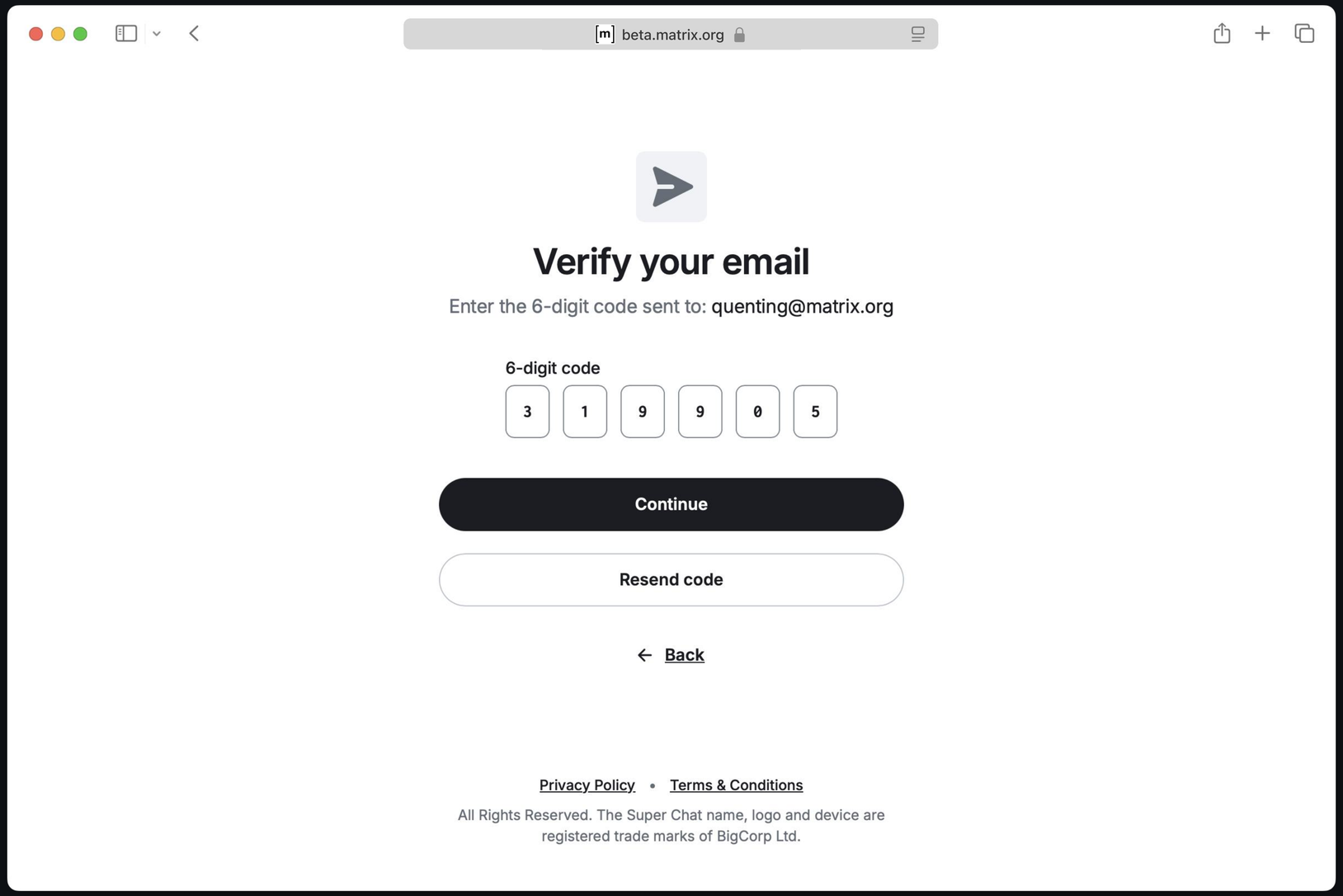

beta.matrix.org
quenting@matrix.org
Password
••••••••••••••••••••
Confirm password
••••••••••••••••••••
I agree to the Terms and Conditions
Success!
CLOUDFLARE
Privacy • Terms
Continue
Cancel
Already have an account? Sign in instead
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.

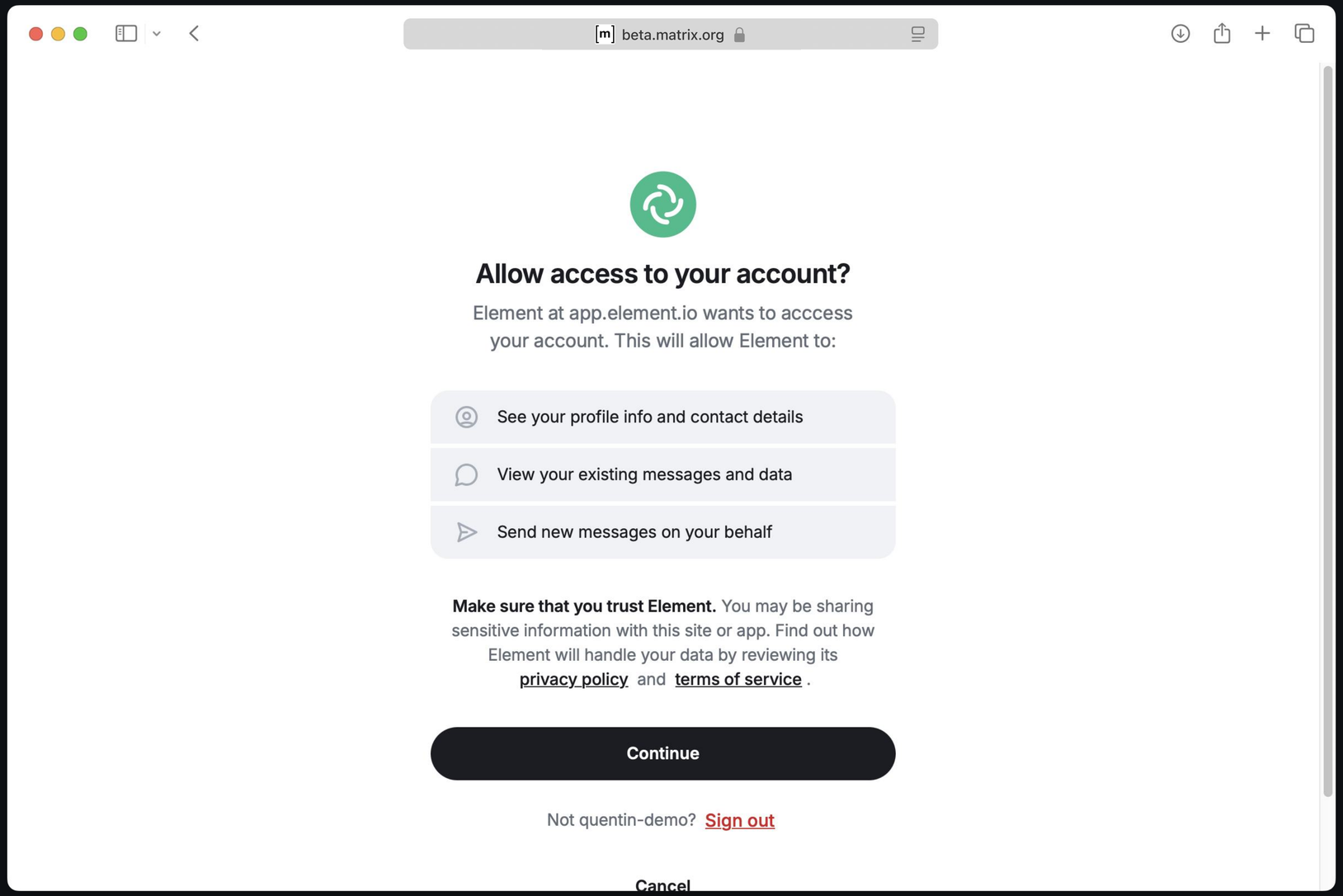

beta.matrix.org
quenting@matrix.org
Password
••••••••••••••••••••
Confirm password
••••••••••••••••••••
I agree to the Terms and Conditions
Success!
CLOUDFLARE
Privacy • Terms
Continue
Cancel
Already have an account? Sign in instead
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.

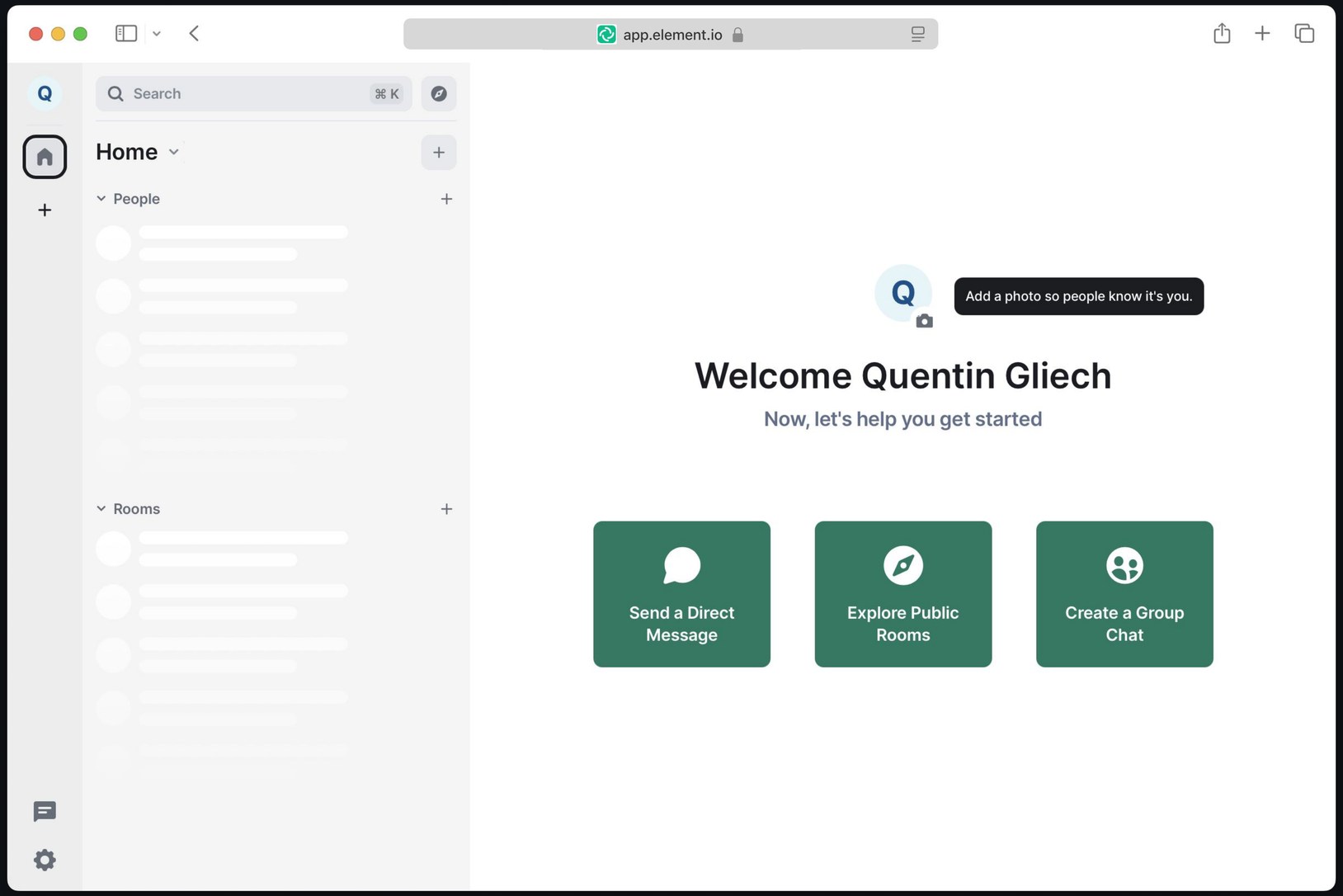

beta.matrix.org
Verify your email
Enter the 6-digit code sent to: quenting@matrix.org
6-digit code
Continue
Resend code
Back
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.


beta.matrix.org
Verify your email
Your friendly authentication server
Your email verification code is: 319905
To: quentin-demo,
Reply-To: No-reply
Inbox - Element
11:48
Hello quentin-demo,
Your verification code to confirm this email address is: 319905


beta.matrix.org
Verify your email
Enter the 6-digit code sent to: quenting@matrix.org
6-digit code
Continue
Resend code
Back
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.


beta.matrix.org
Verify your email
Enter the 6-digit code sent to: quenting@matrix.org
6-digit code
3
1
9
9
0
5
Continue
Resend code
Back
Privacy Policy
Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.


beta.matrix.org
Allow access to your account?
Element at app.element.io wants to acccess your account. This will allow Element to:
See your profile info and contact details
View your existing messages and data
Send new messages on your behalf
Make sure that you trust Element. You may be sharing sensitive information with this site or app. Find out how Element will handle your data by reviewing its privacy policy and terms of service .
Continue
Not quentin-demo? Sign out
Cancel


app.element.io
Search ⌘ K
Home
People
Rooms
Add a photo so people know it's you.
Welcome Quentin Gliech
Now, let's help you get started
Send a Direct Message
Explore Public Rooms
Create a Group Chat

Issue #1:
Auth steps are dynamic, should be specced and supported by clients

Issue #1:
Auth steps are dynamic, should be specced and supported by clients

Introducing new steps is hard

If we want a nice user experience, **client should natively support** all the steps presented. This makes it **harder to introduce new mechanisms**, and raises the bar for new client implementations

Issue #1:
Auth steps are dynamic, should be specced and supported by clients

Introducing new steps is hard

If we want a nice user experience, **client should natively support** all the steps presented. This makes it **harder to introduce new mechanisms**, and raises the bar for new client implementations

Fallback mechanism isn’t enough

UIA has a fallback mechanism for making new steps work, but they are **not well designed for mobile**, and don’t provide a coherent design for clients

Issue #1:
Auth steps are dynamic, should be specced and supported by clients

Introducing new steps is hard

If we want a nice user experience, **client should natively support** all the steps presented. This makes it **harder to introduce new mechanisms**, and raises the bar for new client implementations

Fallback mechanism isn’t enough

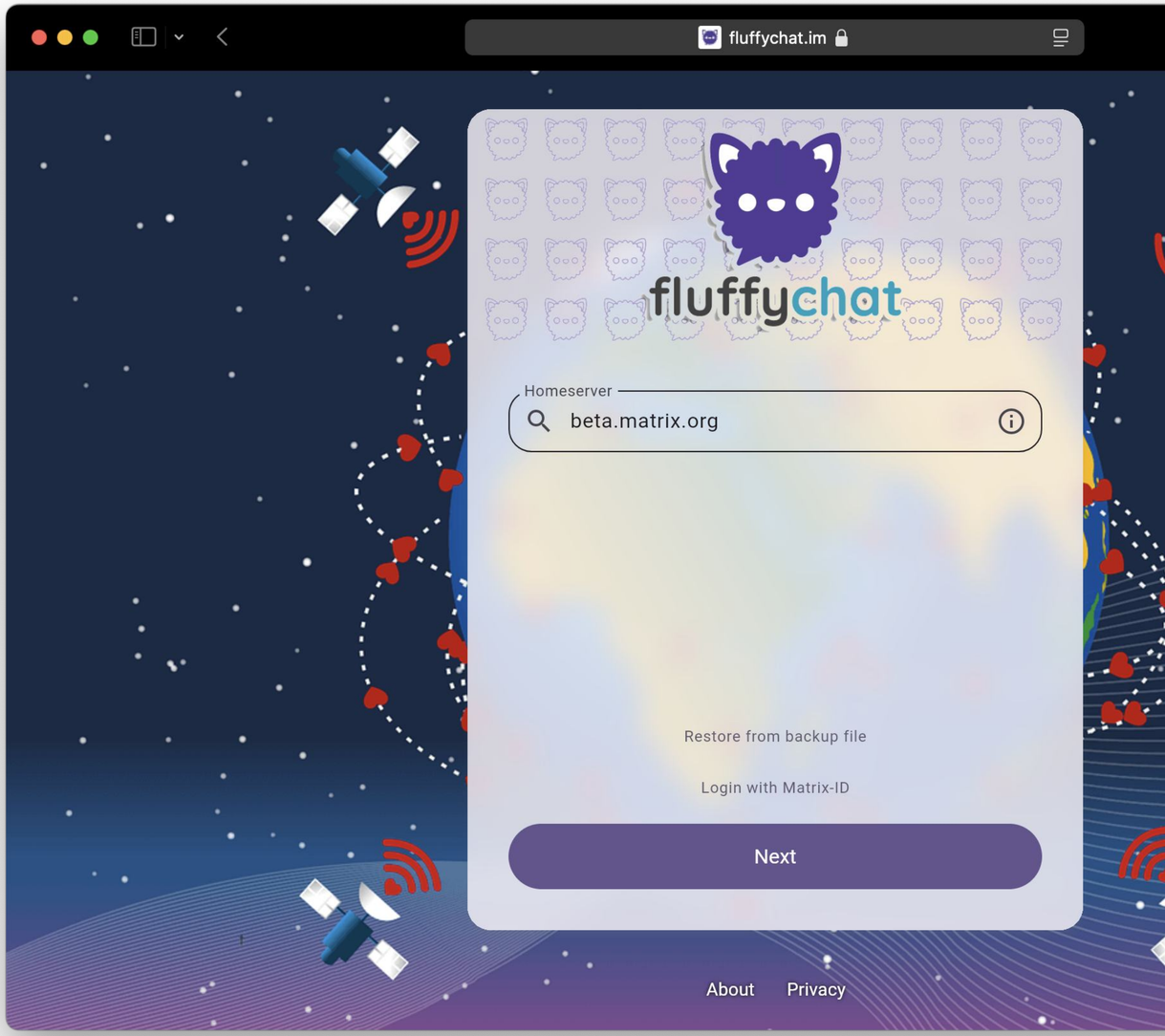
UIA has a fallback mechanism for making new steps work, but they are **not well designed for mobile**, and don’t provide a coherent design for clients

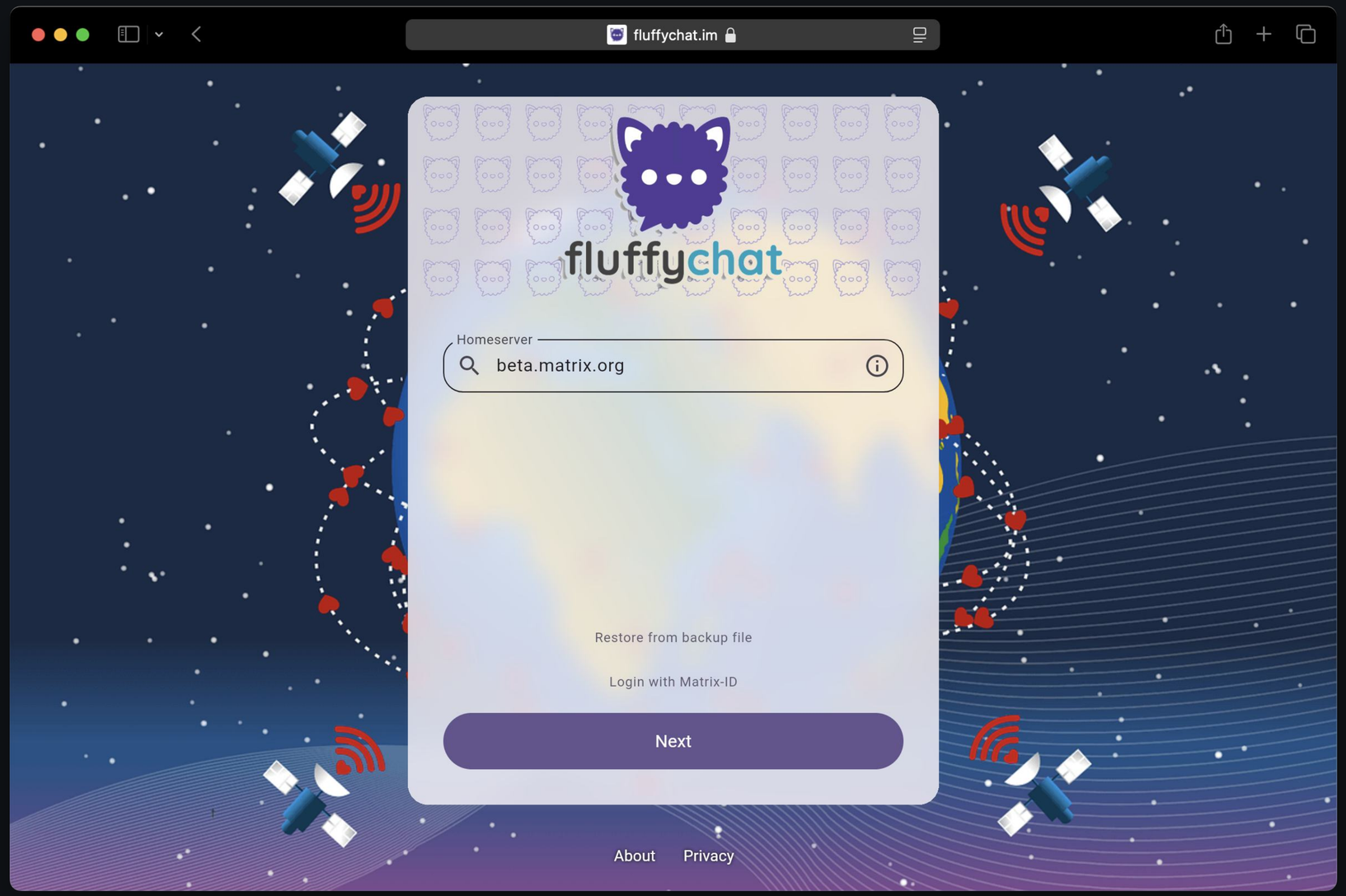
Dynamic user-interfaces are hard

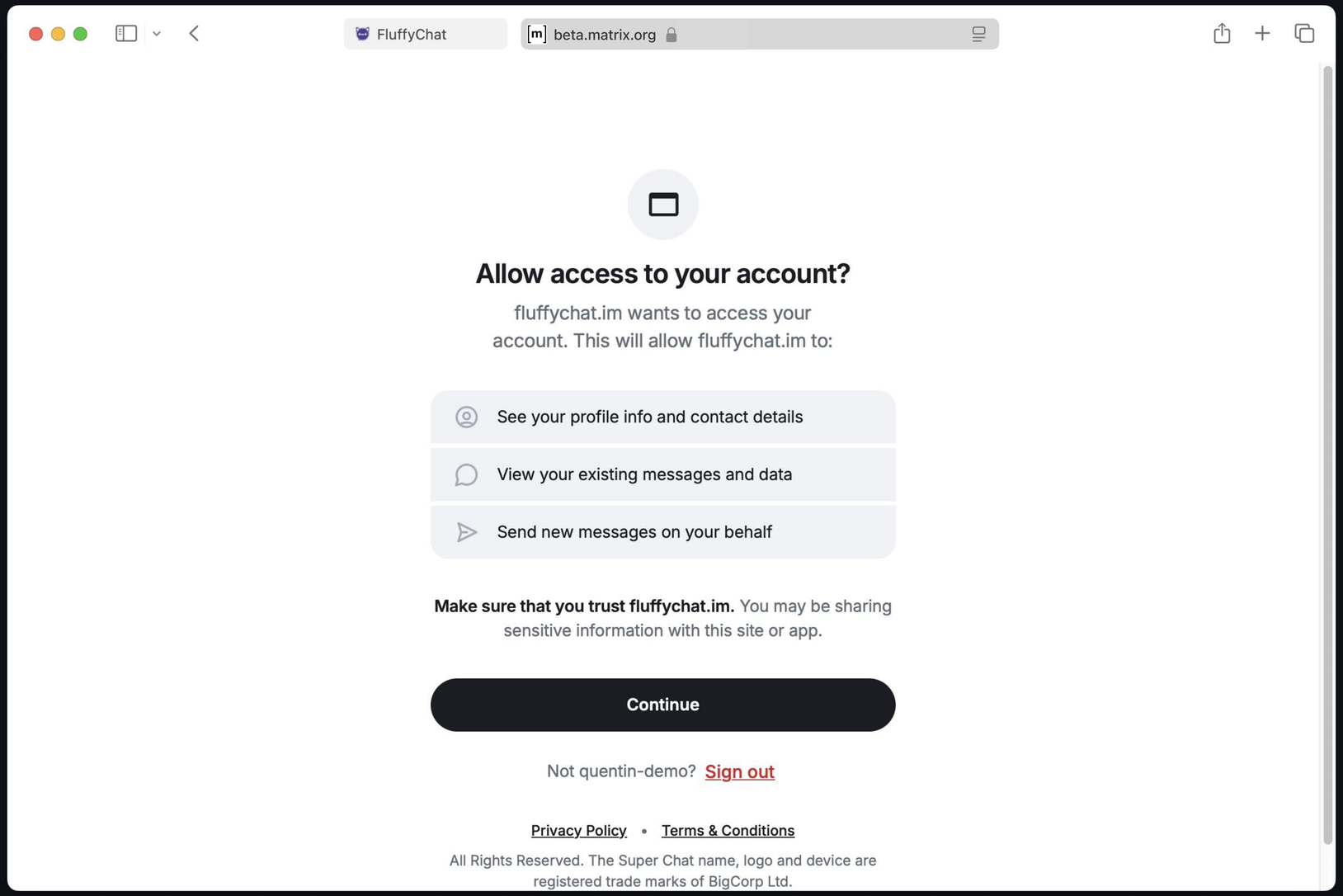
The **list of steps** presented in UIA is **dynamic**, which makes it hard for clients to build a coherent flow if they can’t predict what combination is going to be asked by the server

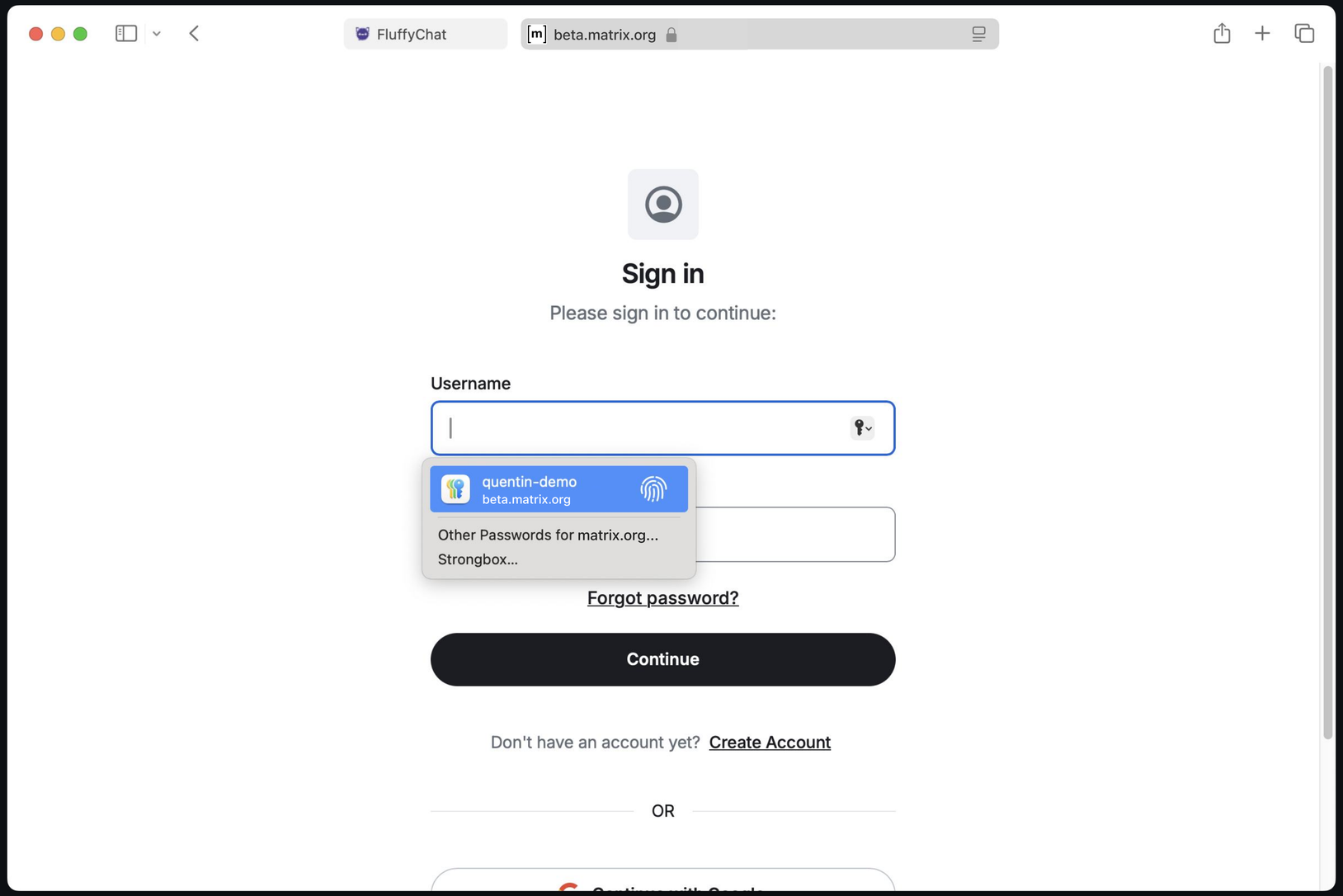
Let’s login in another client

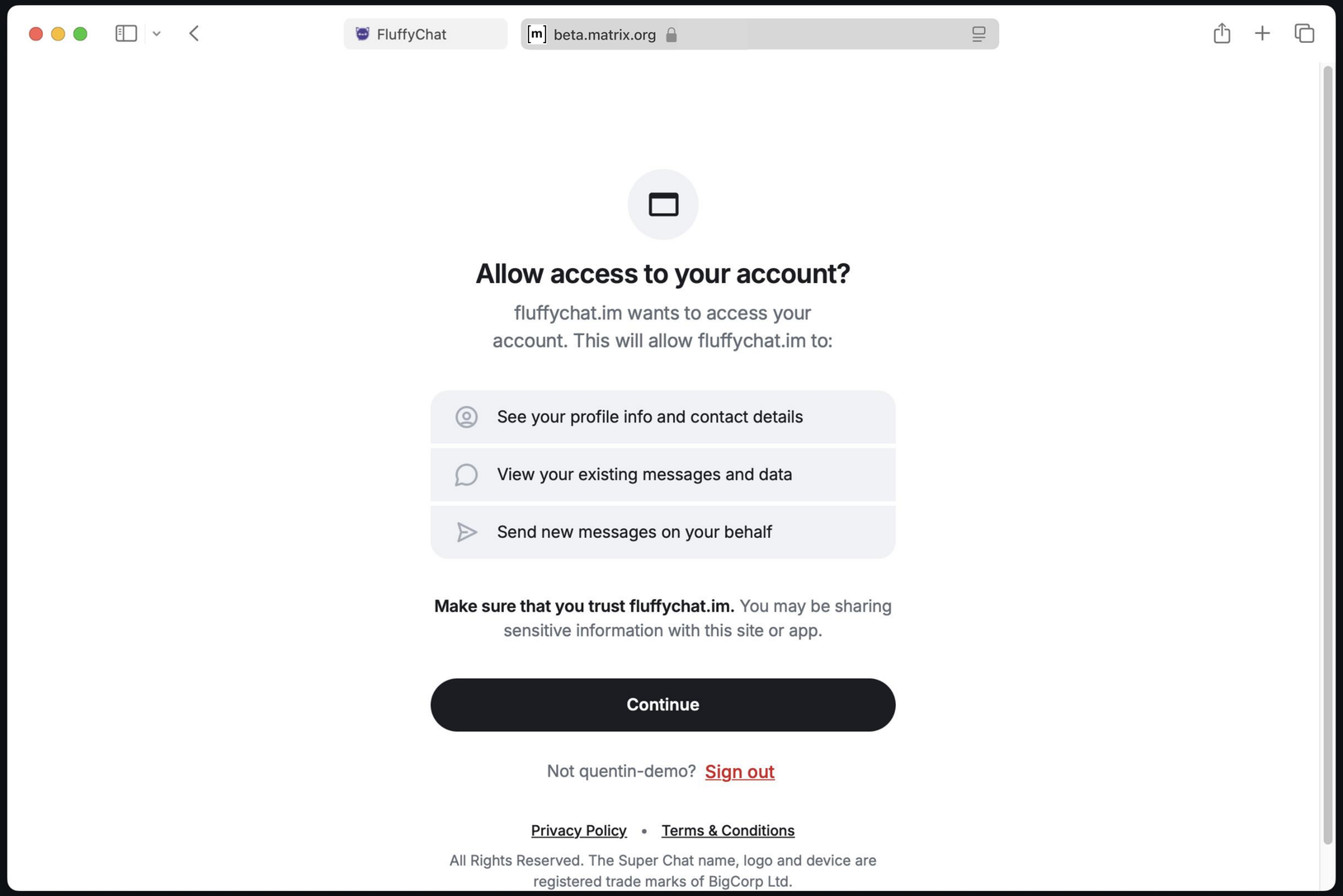
I would like to try out FluffyChat
with my brand new Matrix account

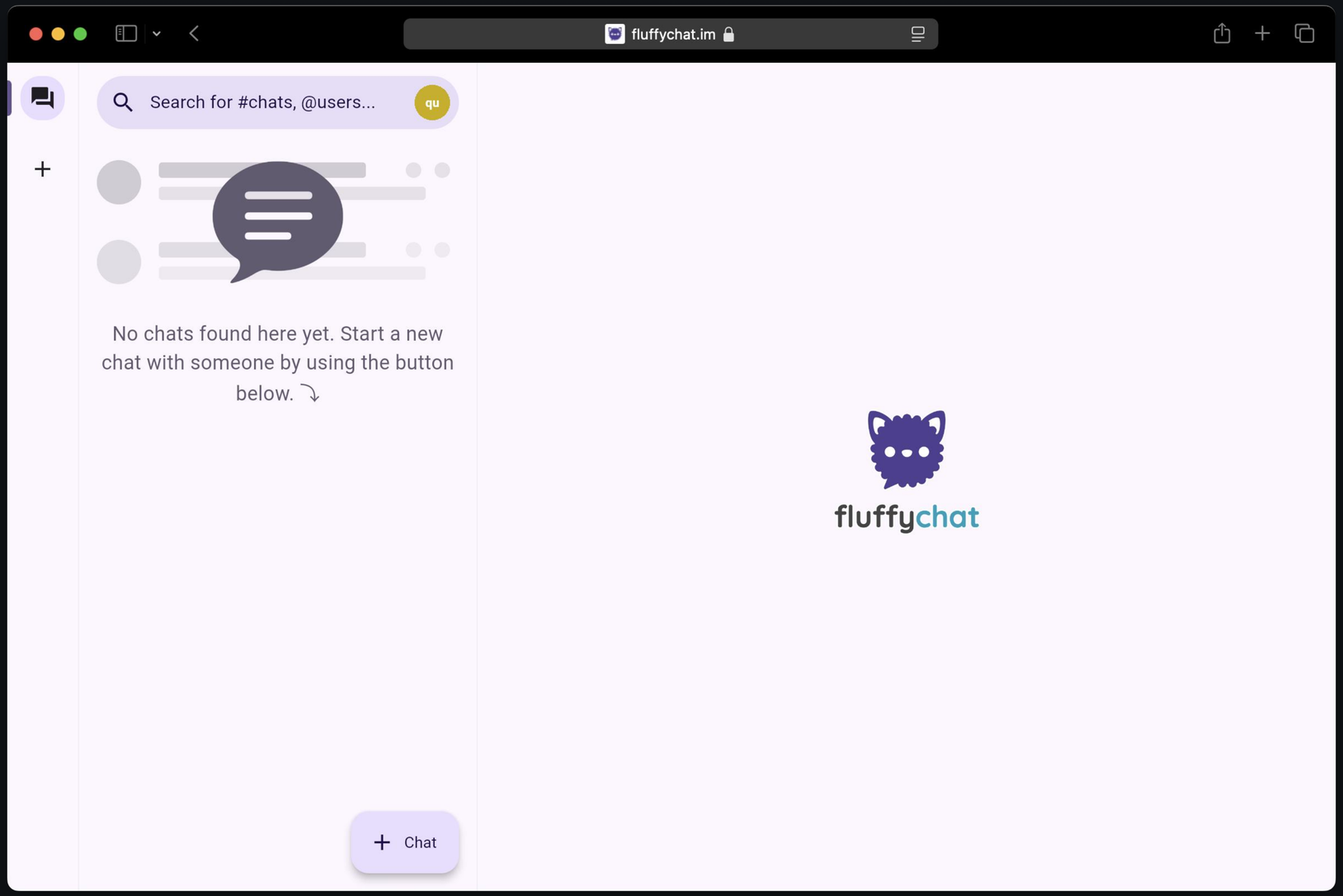



fluffychat.im
fluffychat
Homeserver
beta.matrix.org
Restore from backup file
Login with Matrix-ID
Next
About
Privacy


FluffyChat
beta.matrix.org
Allow access to your account?
fluffychat.im wants to access your account. This will allow fluffychat.im to:
See your profile info and contact details
View your existing messages and data
Send new messages on your behalf
Make sure that you trust fluffychat.im. You may be sharing sensitive information with this site or app.
Continue
Not quentin-demo? Sign out
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.

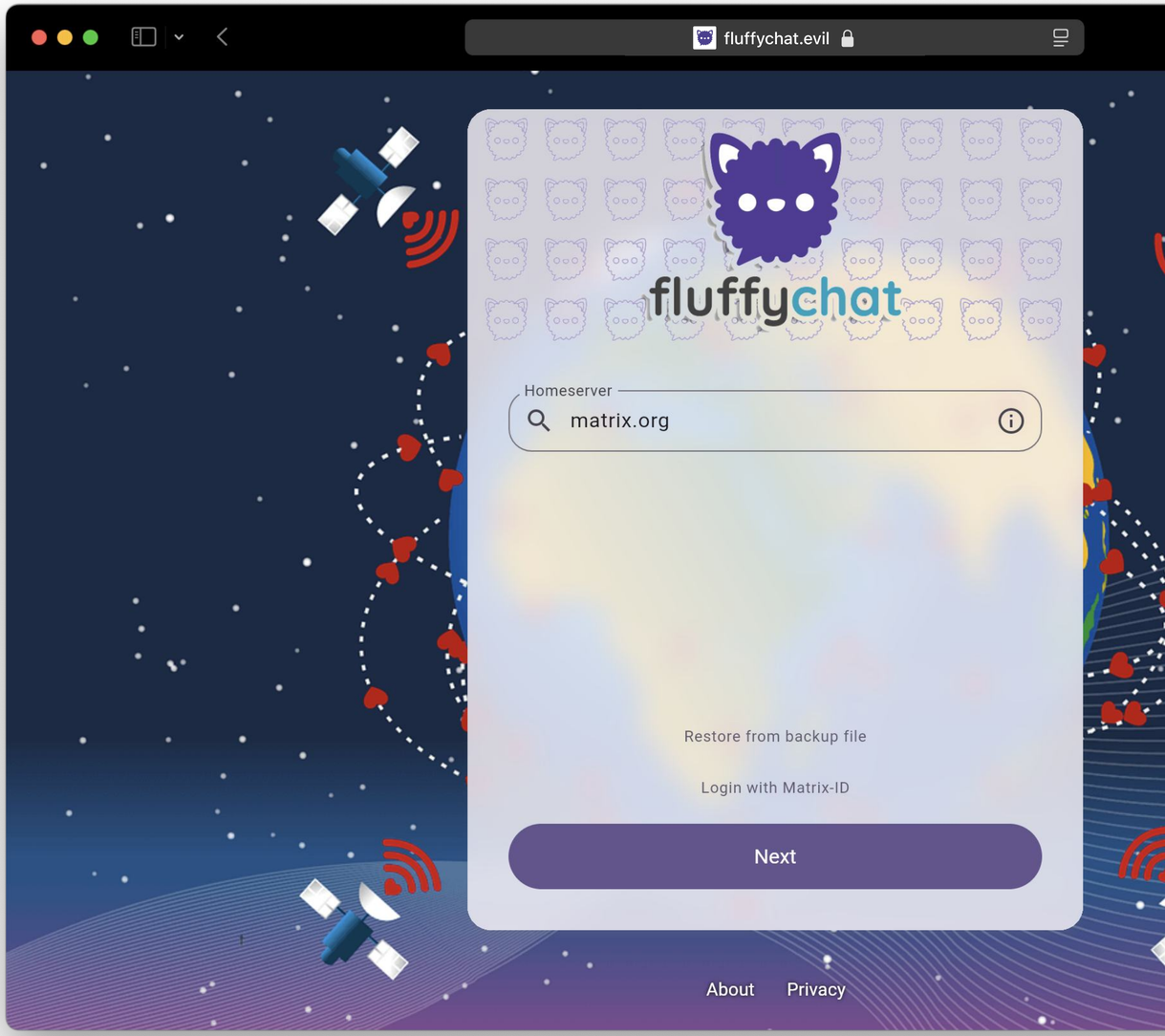

FluffyChat
beta.matrix.org
Sign in
Please sign in to continue:
Username
quentin-demo
beta.matrix.org
Other Passwords for matrix.org...
Strongbox...
Forgot password?
Continue
Don't have an account yet? Create Account
OR

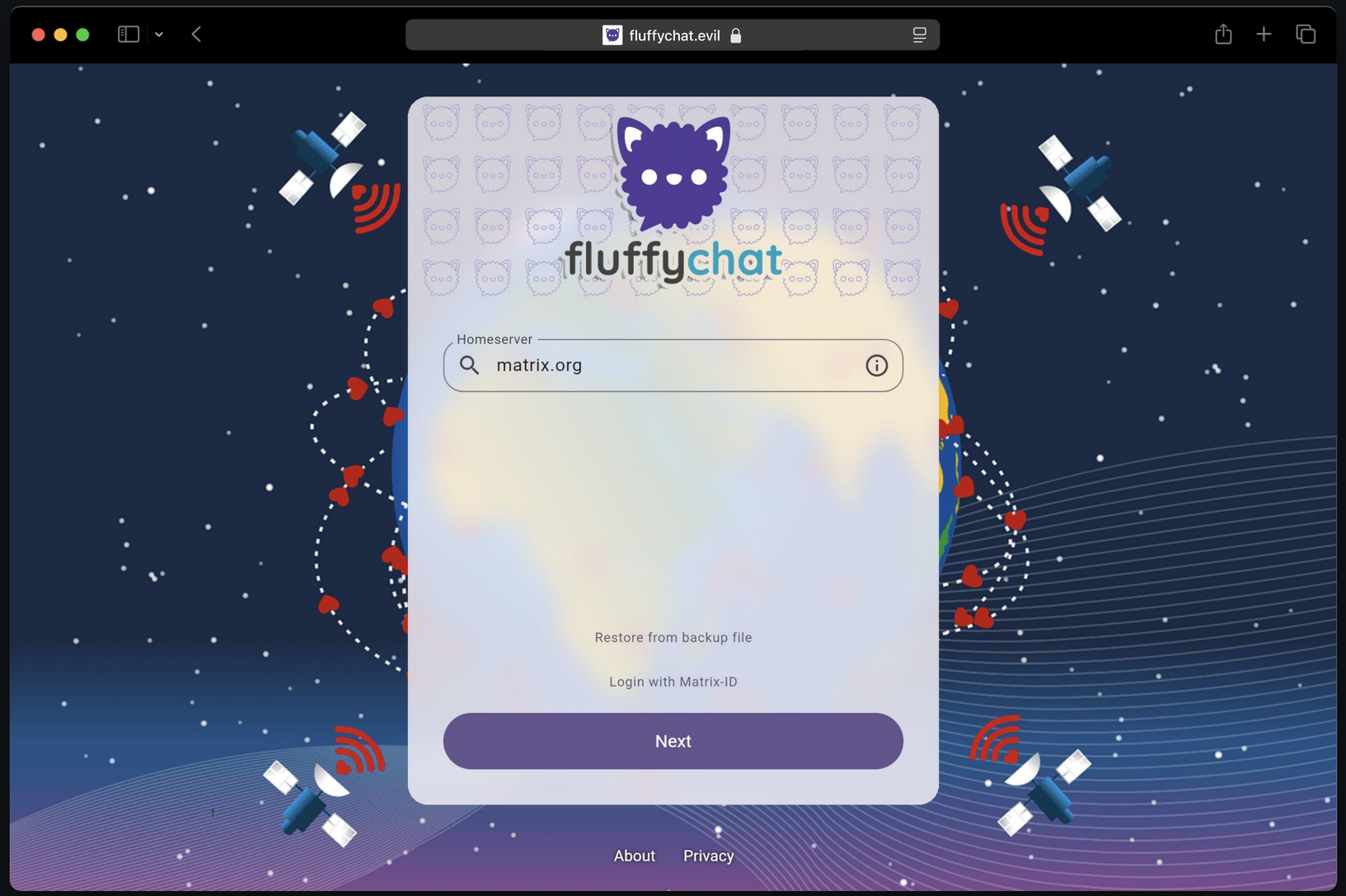

FluffyChat
beta.matrix.org
Allow access to your account?
fluffychat.im wants to access your account. This will allow fluffychat.im to:
See your profile info and contact details
View your existing messages and data
Send new messages on your behalf
Make sure that you trust fluffychat.im. You may be sharing sensitive information with this site or app.
Continue
Not quentin-demo? Sign out
Privacy Policy • Terms & Conditions
All Rights Reserved. The Super Chat name, logo and device are registered trade marks of BigCorp Ltd.

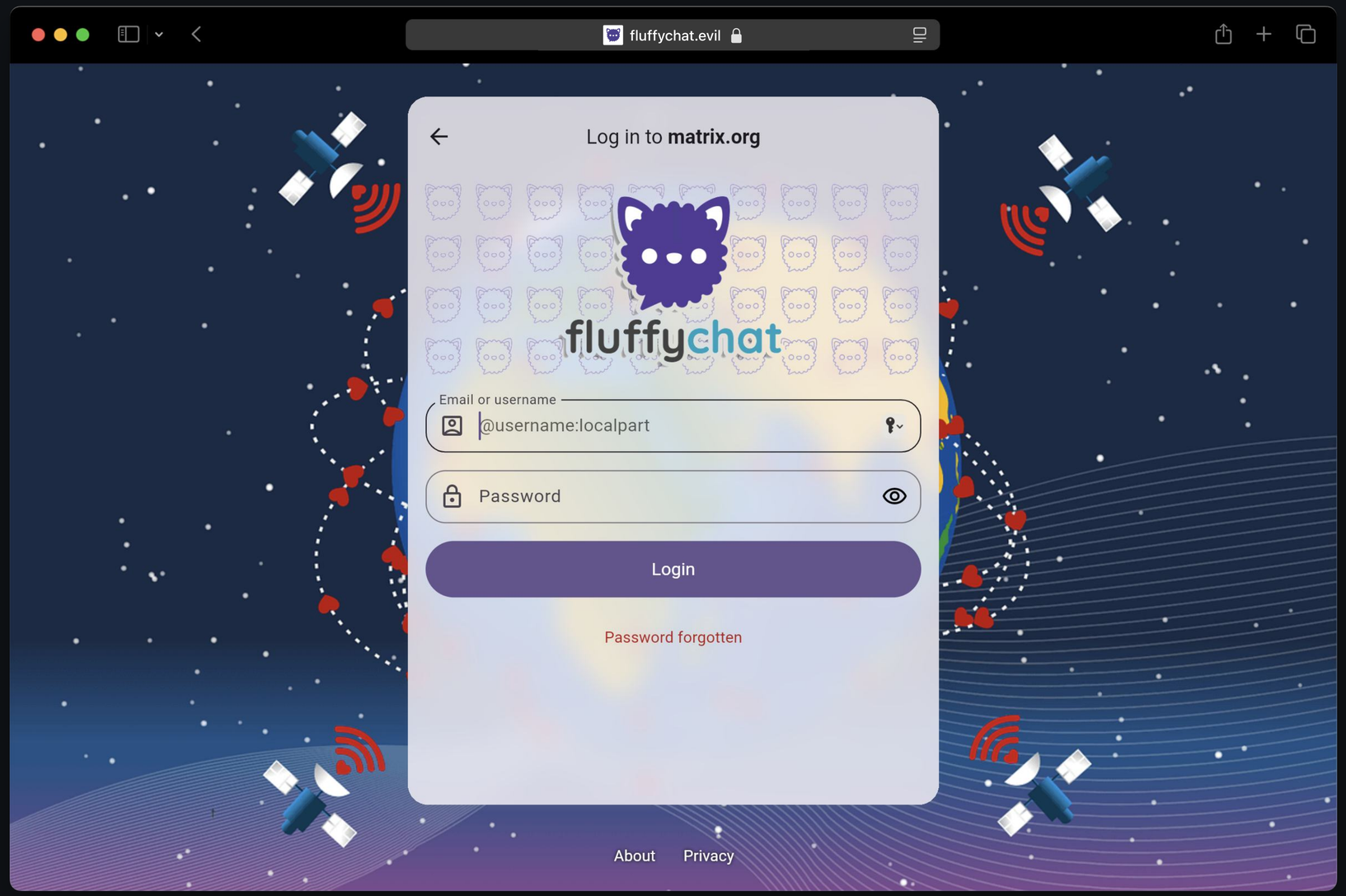

fluffychat.im
Search for #chats, @users...
qu
No chats found here yet. Start a new chat with someone by using the button below.
Chat
fluffychat

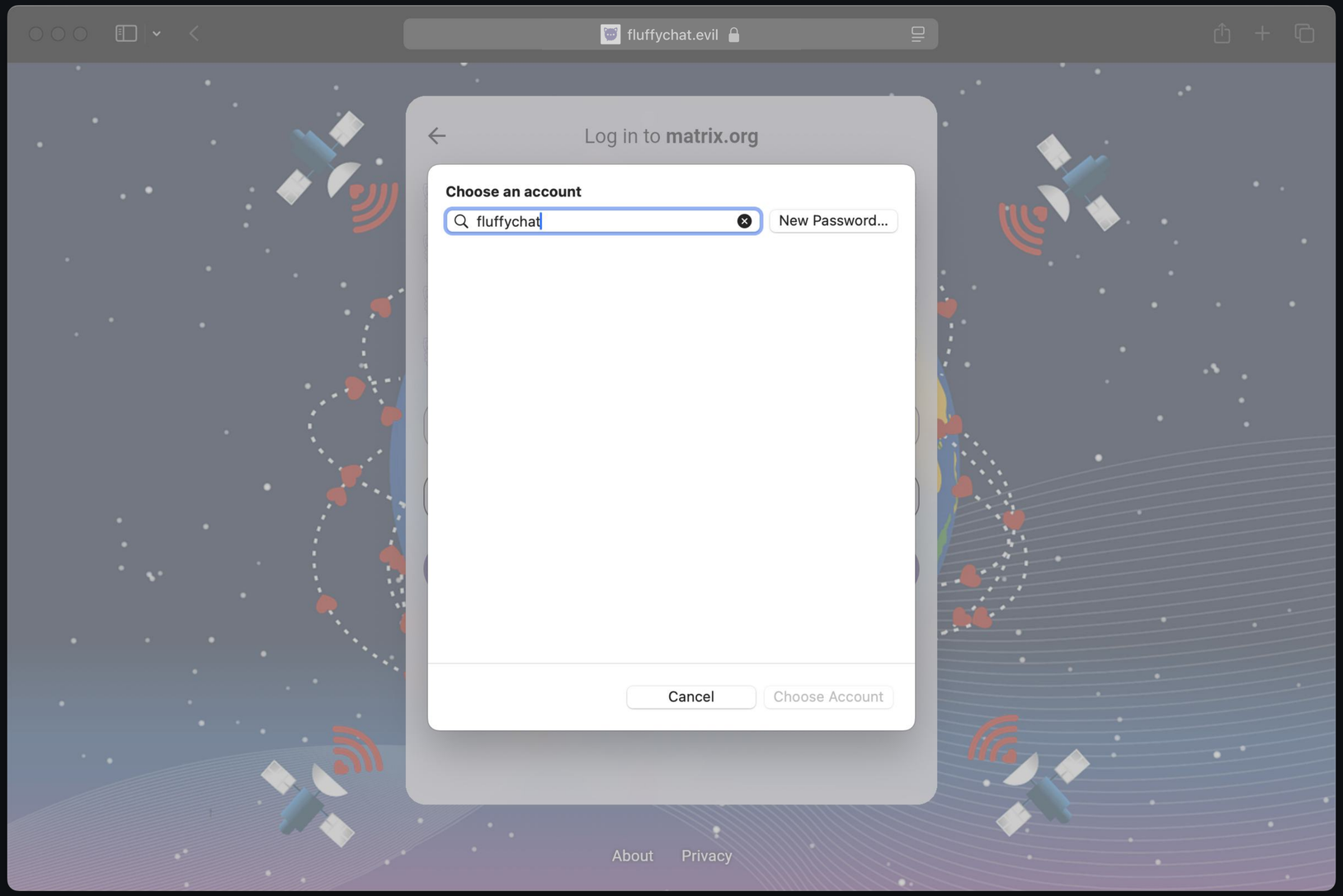
And on Matrix.org?

How does a login look like on the current authentication stack?




fluffychat.evil
fluffychat
Homeserver
matrix.org
Restore from backup file
Login with Matrix-ID
Next
About
Privacy


fluffychat.evil
Log in to matrix.org
fluffychat
Email or username
@username:localpart
Password
Login
Password forgotten
About
Privacy

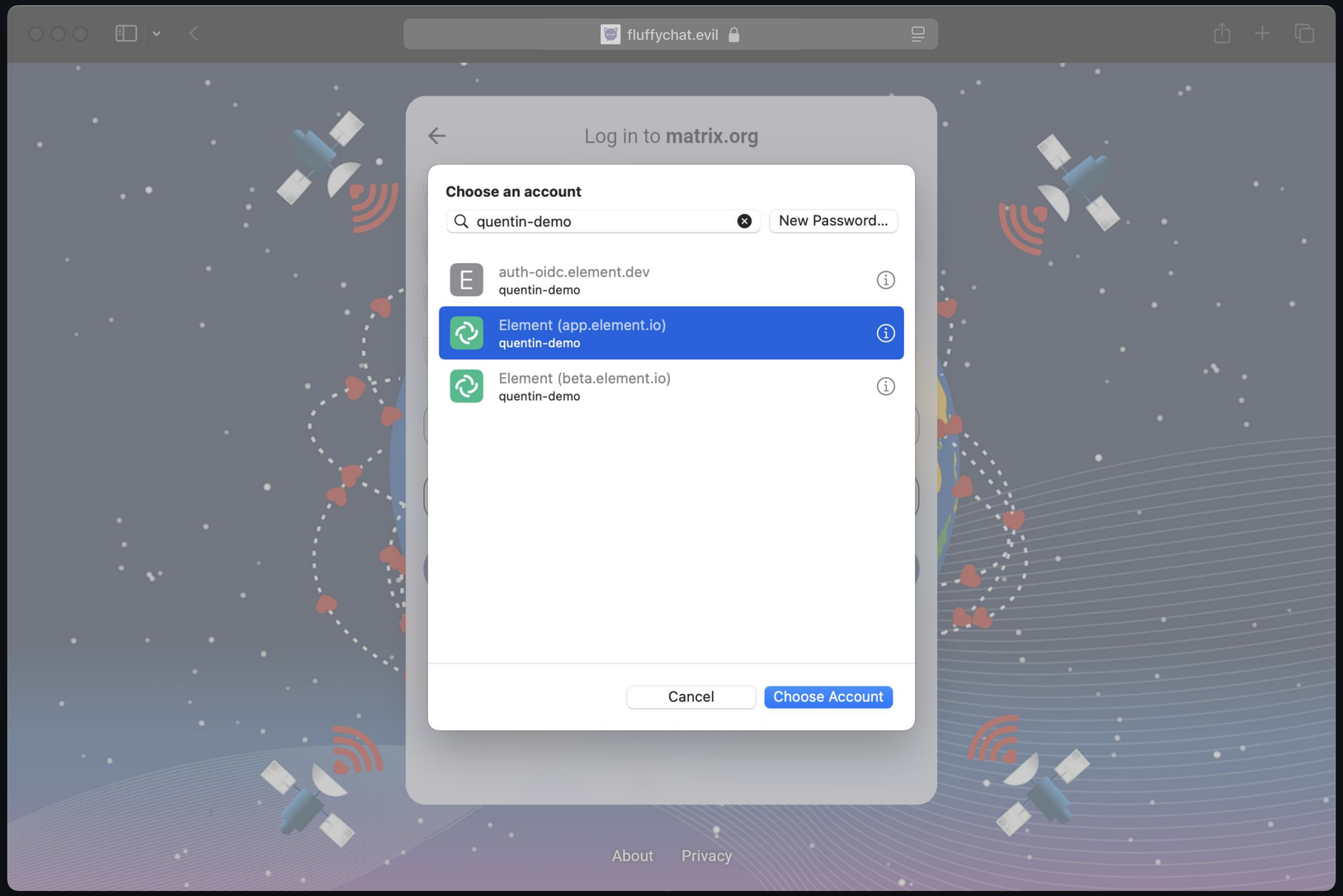

fluffychat.evil
Log in to matrix.org
Choose an account
fluffychat
New Password...
Cancel
Choose Account
About
Privacy

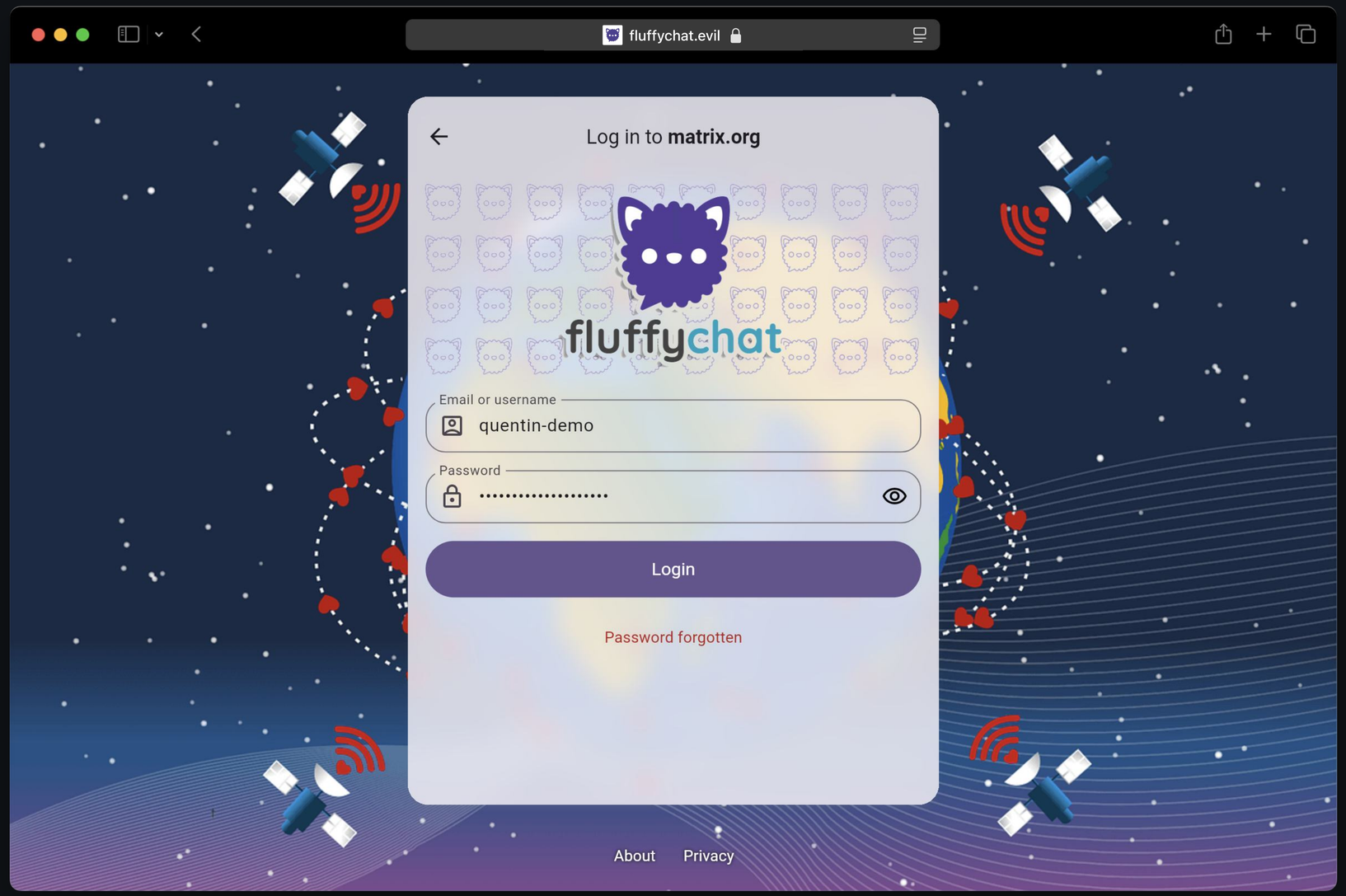
Issue #2: Credentials are bound to the domain

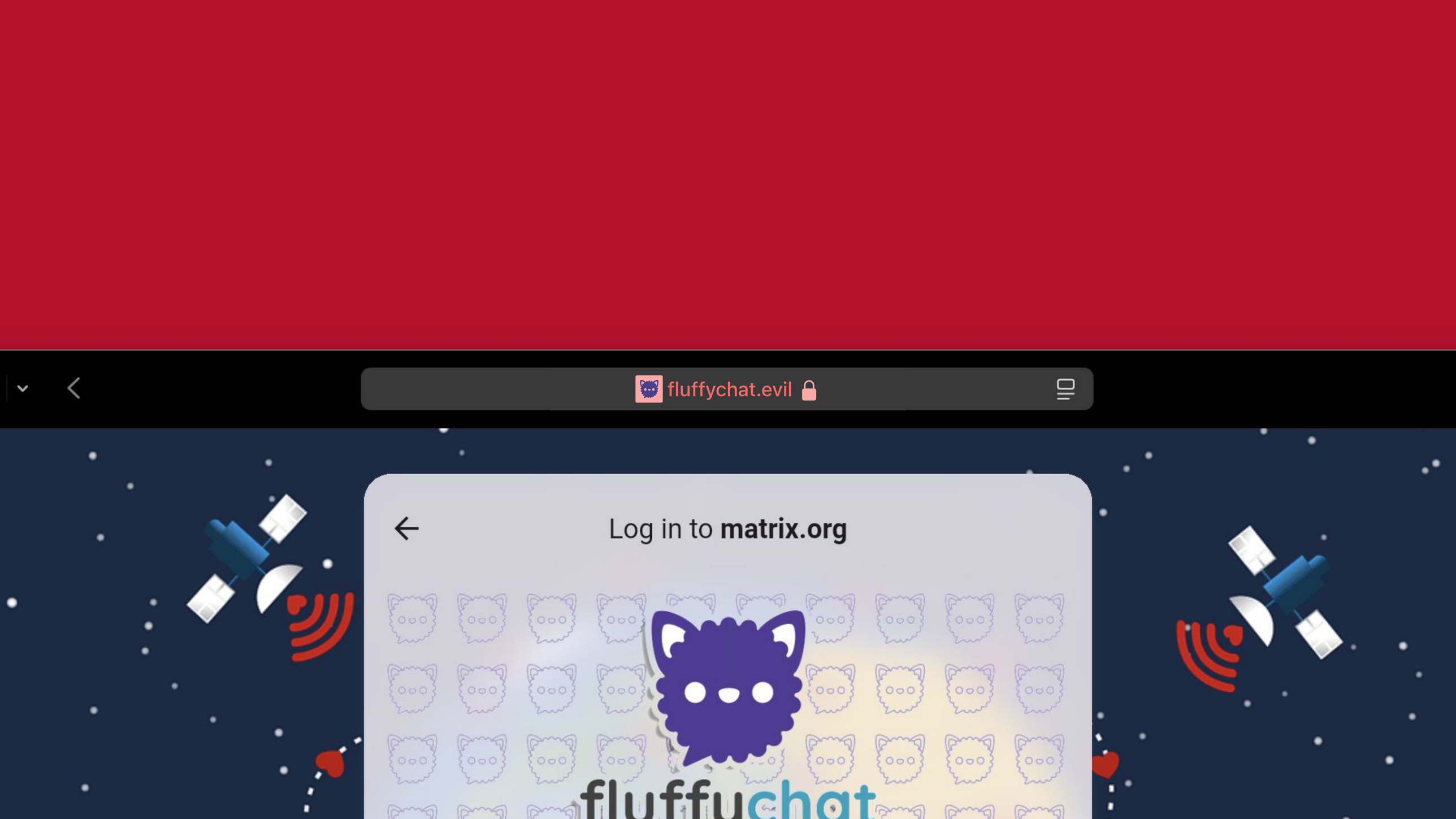
Your credentials are supposed to be bound to the **service** and **not the client**

Password managers don’t behave well

CAPTCHAs, Passkeys, client certificates are all **bound to the domain**


fluffychat.evil
Log in to matrix.org
Choose an account
quentin-demo
New Password...
auth-oidc.element.dev
quentin-demo
Element (app.element.io)
quentin-demo
Element (beta.element.io)
quentin-demo
Cancel
Choose Account
About
Privacy


fluffychat.evil
Log in to matrix.org
fluffychat
Email or username
quentin-demo
Password
Login
Password forgotten
About
Privacy


fluffychat.evil
Log in to matrix.org
fluffychat

Issue #3:
Clients have access to the full credentials

Issue #3:
Clients have access to the full credentials

Are destructive actions really protected?

Deactivating the account, changing **email addresses**, changing **password** are supposed to be protected by UIA.
On matrix.org, that means **re-submitting the password**

Issue #3:
Clients have access to the full credentials

Are destructive actions really protected?

Deactivating the account, changing **email addresses**, changing **password** are supposed to be protected by UIA.
On matrix.org, that means **re-submitting the password**

Client could have leaked or saved the credentials

Users are currently handing out their credentials to many parties, effectively **widening the attack surface**

Issue #3:
Clients have access to the full credentials

Are destructive actions really protected?

Deactivating the account, changing **email addresses**, changing **password** are supposed to be protected by UIA.
On matrix.org, that means **re-submitting the password**

Client could have leaked or saved the credentials

Users are currently handing out their credentials to many parties, effectively **widening the attack surface**

Can’t restrict access to the client

If the client has full access over the credentials, we can’t design anything that gives restricted access to a client.
We cannot give partial access to the account, to a subset of rooms, to a specific room type, etc.

Issue #1:
Auth steps
are dynamic

Issue #2:
Credentials are
bound to the domain

Issue #3:
Clients have access
to the full credentials

Issue #1:
Auth steps
are dynamic

Issue #2:
Credentials are
bound to the domain

Issue #3:
Clients have access
to the full credentials

Auth through the browser fixes this

The server can present any kind of browser-based UI, in a streamlined and coherent way

Issue #1: Auth steps are dynamic

Credentials are bound to the **domain of the service** instead of the domain of the client

Issue #2: Credentials are bound to the domain

Clients don’t see the full credentials. The user enters them in a UI controlled by the service

Issue #3: Clients have access to the full credentials

Matrix already has m.login.sso

1. The client redirects to the homeserver to start the authentication
2. The homeserver authenticates the user
3. It redirects back to the client with a single-use code
4. The client exchanges that code to get an access token

This mechanism is simple yet powerful

Me : mom can we have **Secure auth through the browser** ?

Mom : no, we have **Secure auth through the browser** at home

at home : `m.login.sso`

m.login.sso is a ~~bad~~ limited version of the OAuth 2.0 authorisation code grant

OAuth 2.0

The industry-standard authorisation **framework**, developed within the **IETF** OAuth Working Group

OpenID Connect

An identity layer **on top of OAuth 2.0**, helping with better interoperability, developed by the **OpenID Foundation**

OAuth 2.0

The industry-standard authorisation **framework**, developed within the **IETF** OAuth Working Group

OpenID Connect

An identity layer **on top of OAuth 2.0**, helping with better interoperability, developed by the **OpenID Foundation**

They are building blocks

We need to define a *profile* on top of them:

- what homeservers and clients **must** support to be compliant
- how we model Matrix concepts with them

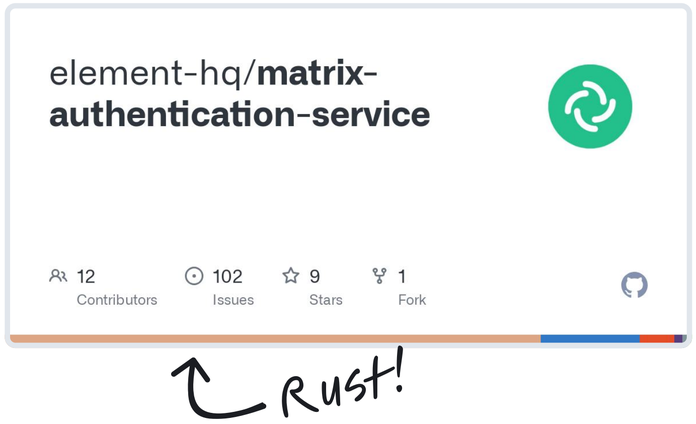
Cover many use cases

- browser-based flow with the **authorisation code grant**
- authorise on another device with the **device code grant**
- “service account” use cases with the **client credentials grant**

Introducing matrix-authentication-service

Or: how 2940 commits and 80k lines of code later, we fixed auth.

Introducing matrix-authentication-service



Rewrite from the ground-up

Complete rewrite of Synapse's auth logic
Focused on a **good auth UX**
Could be **used by other** homeservers

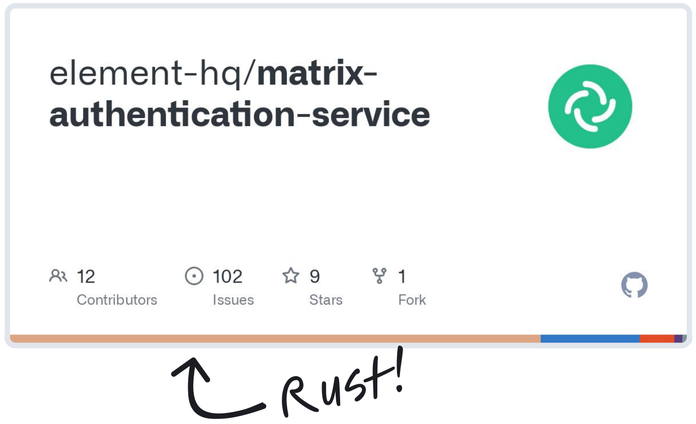
Main flow is browser-based

Even for local-password authentication, making **password managers 'just work'**
Registration flow is ✨ delightful ✨

Account management UI

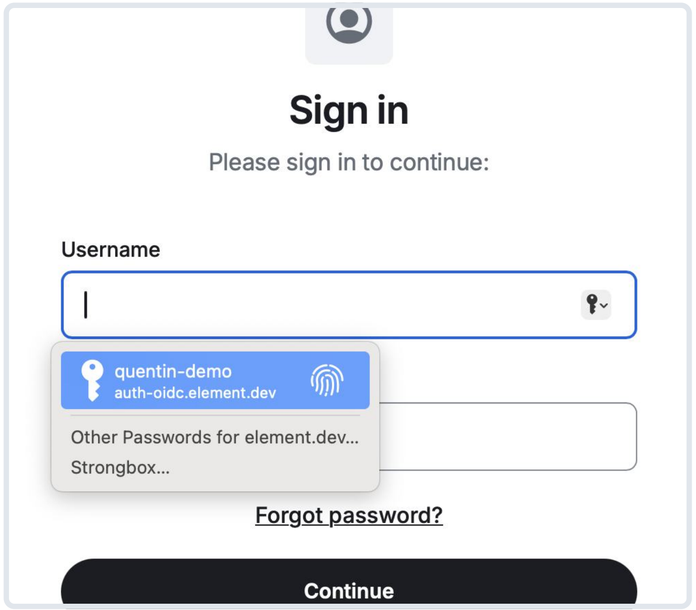
Dedicated **UI for managing your sessions**, password, email addresses, etc.

Introducing matrix-authentication-service



Rewrite from the ground-up

Complete rewrite of Synapse's auth logic
Focused on a **good auth UX**
Could be **used by other** homeservers



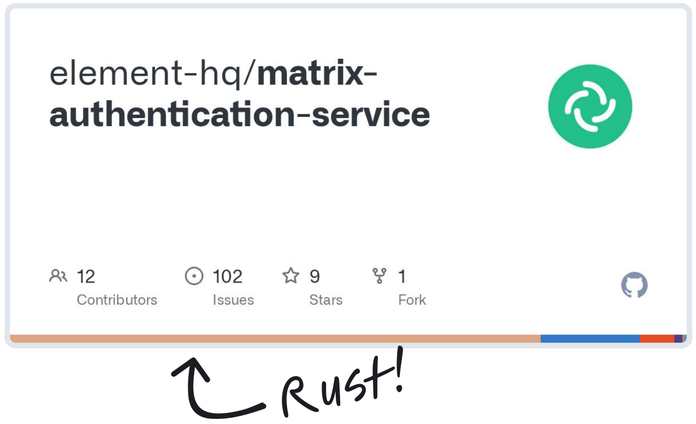
Main flow is browser-based

Even for local-password authentication, making **password managers 'just work'**
Registration flow is ✨ delightful ✨

Account management UI

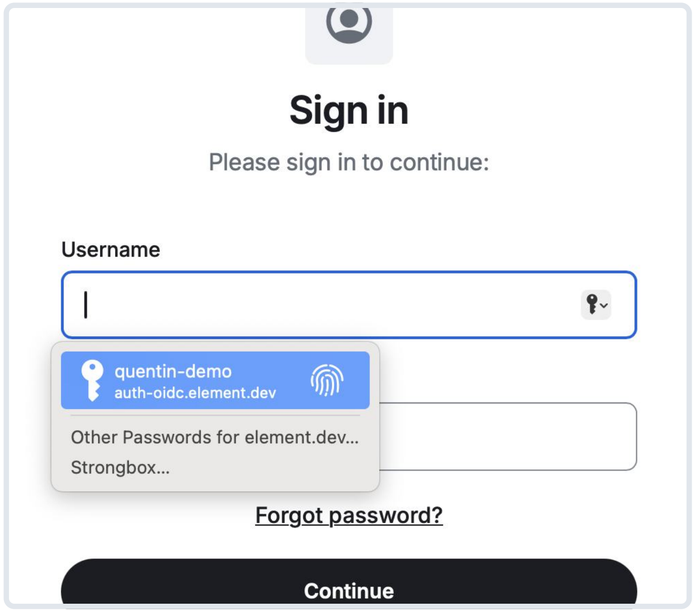
Dedicated **UI for managing your sessions**, password, email addresses, etc.

Introducing matrix-authentication-service



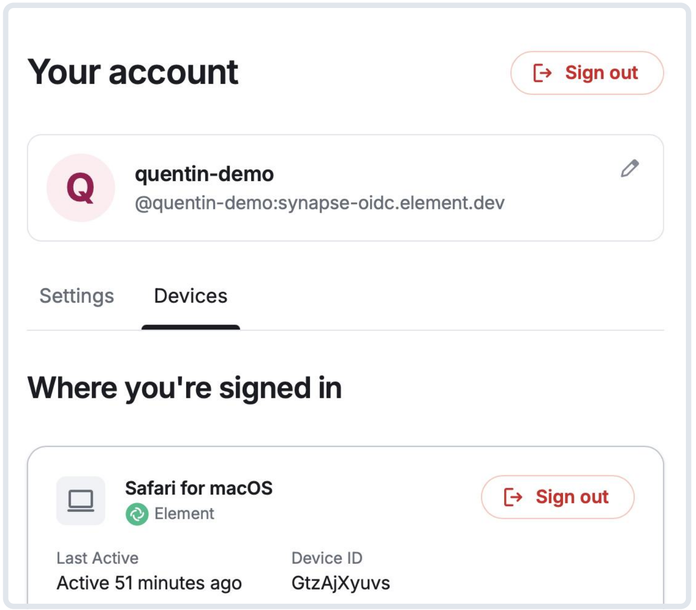
Rewrite from the ground-up

Complete rewrite of Synapse's auth logic
Focused on a **good auth UX**
Could be **used by other** homeservers



Main flow is browser-based

Even for local-password authentication, making **password managers 'just work'**
Registration flow is ✨ delightful ✨

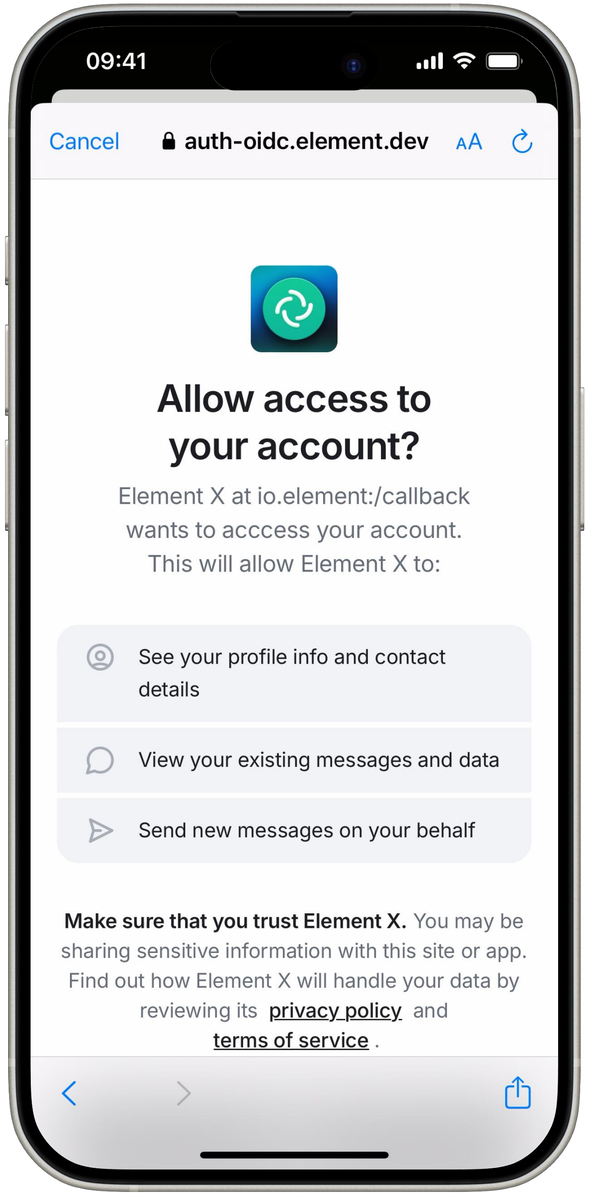


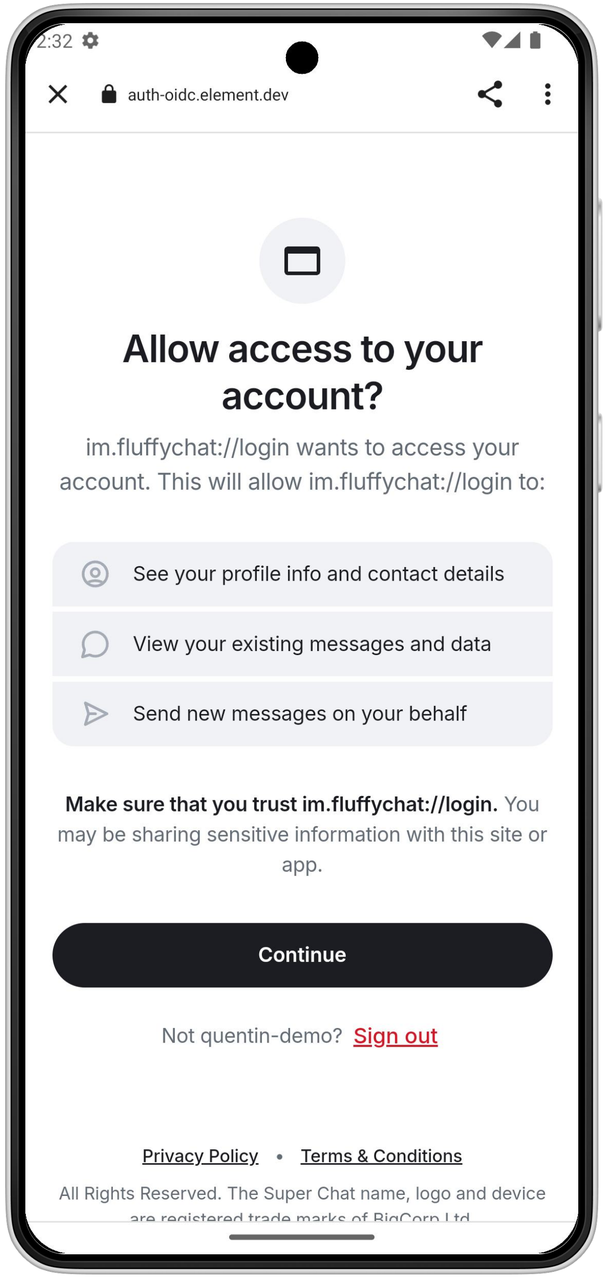
Account management UI

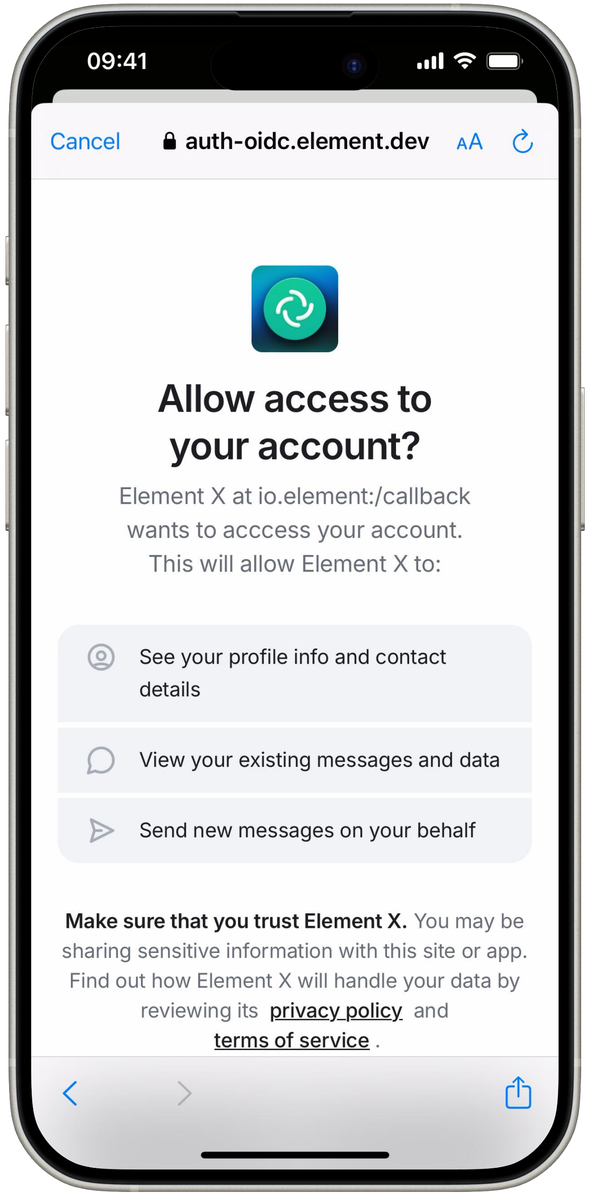
Dedicated **UI for managing your sessions**, password, email addresses, etc.

Works with most clients out of the box

Supports both **m.login.sso**
and the new **OAuth 2.0-based** flows





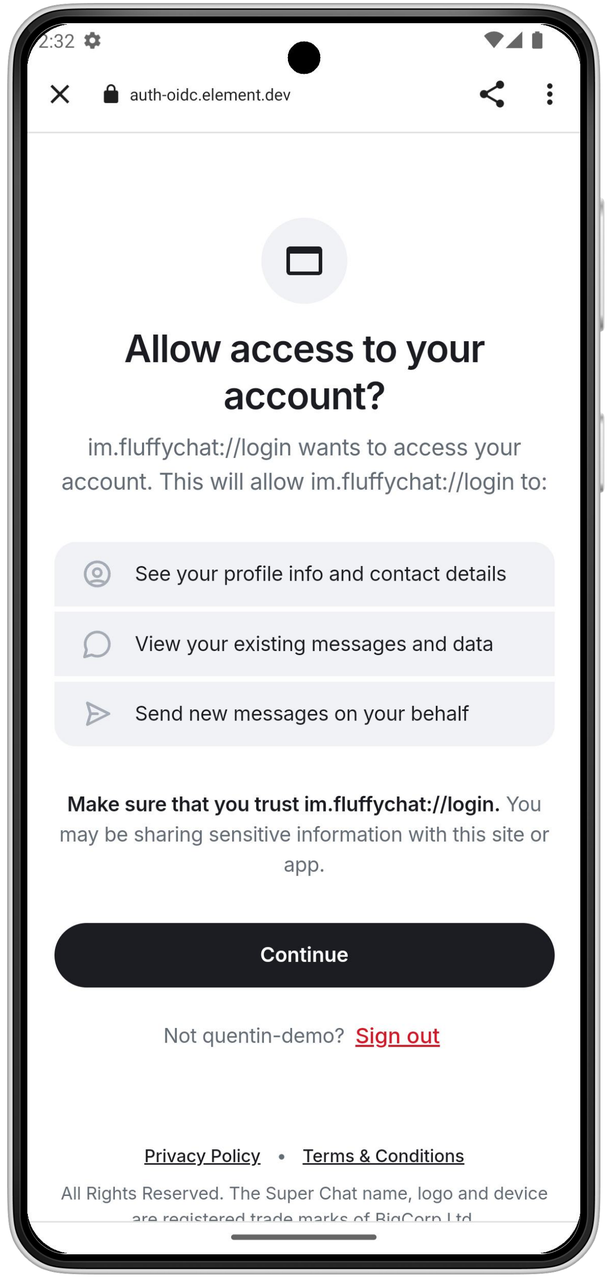


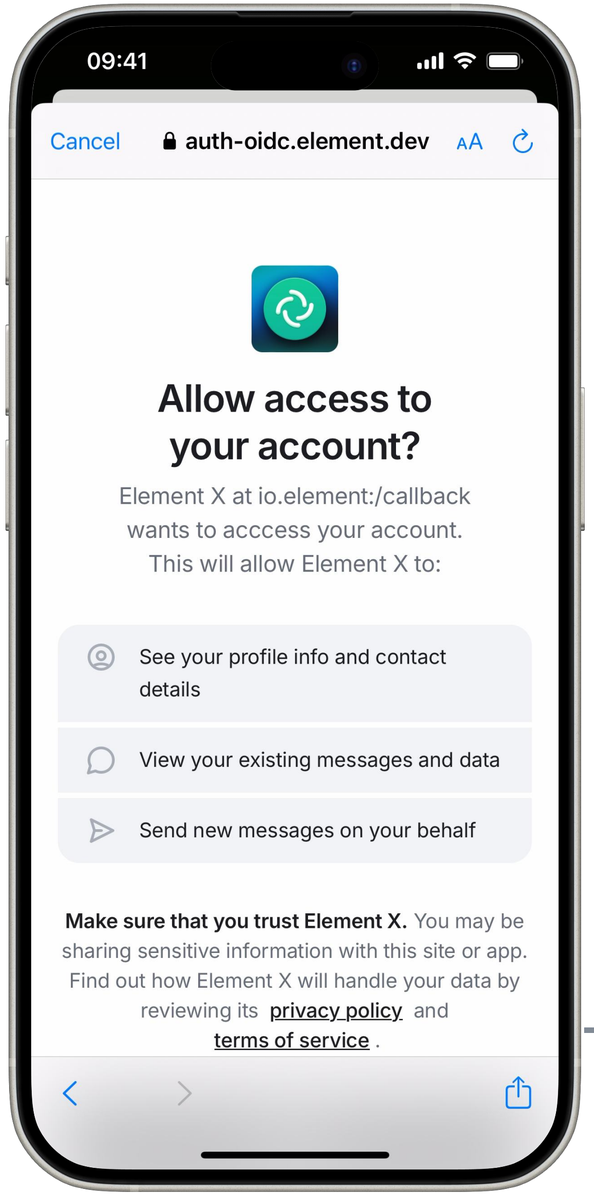
OAuth 2.0

Redirect URI

Access granted

m.login.sso





OAuth 2.0

m.login.sso

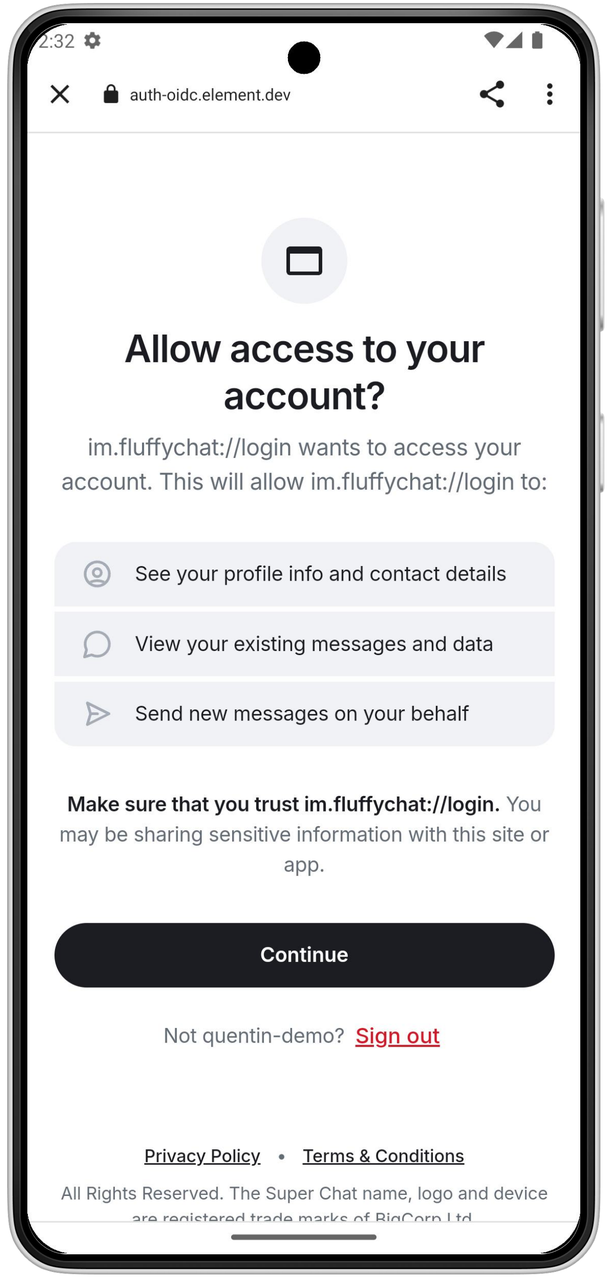
Logo

Redirect URI

Name

Access granted

Privacy policy and terms of service



Protocol improvements over legacy auth

Protocol improvements over legacy auth

Regarding user-experience

- Rich **client metadata**
- Can return to the client **with an error**
- Can initiate the flow with **different intents**

Protocol improvements over legacy auth

Regarding user-experience

- Rich **client metadata**
- Can return to the client **with an error**
- Can initiate the flow with **different intents**

Regarding security

- **Strict redirect** URL checks
- Protection against **redirect hijacks**
- **Scoped** access tokens
- **Short-lived** access tokens

Key takeaways

What protocol changes we're proposing

Authentication through the browser makes the most sense

Many credentials are domain bound & having each authentication step handled by the client doesn't scale well

We're moving Matrix to use an OAuth 2.0/OIDC-based auth system

UIA was a good idea, but too complex in practice.
OAuth 2.0 gives us mature building blocks to replace it.

Matrix isn't an account-management and authentication protocol

It currently takes a significant portion of the spec, even though it is not the core of the project

Key takeaways

On matrix-authentication-service

It is a rewrite from the ground-up

In addition to the better, more modern and secure APIs,
we thought of a better user-experience from the beginning

It replaces part of Synapse

It's not *'just'* an OpenID Connect/OAuth 2.0 provider, but
really the rewrite of Synapse's internal auth system

We'll gradually make it easier to run

The long-term goal is to be a part of Synapse, and make the
upgrade as simple as a regular Synapse upgrade

MSC3861
Next-generation auth for Matrix, based on OAuth 2.0/OIDC

MSC3861
Next-generation auth for Matrix

MSC3861
Next-generation auth for Matrix

MSC2964
“how to login”

Defines how clients should leverage OAuth 2.0 authorization grant to gain access to the account

MSC2966
“client tells about itself”

Defines how clients send metadata about themselves, and what we are enforcing

MSC3861
Next-generation auth for Matrix

MSC2964
“how to login”

Defines how clients should leverage OAuth 2.0 authorization grant to gain access to the account

MSC2965
“discover server params”

Servers have different features and endpoints, this uses OIDC Discovery to discover the “auth server metadata”

MSC2966
“client tells about itself”

Defines how clients send metadata about themselves, and what we are enforcing

MSC2967
“base for scoped access”

Opens up the space for scoped access, but only specifies an “access to everything” scope for now

Support the existing ecosystem

MSC4911 to link to account-management pages

Account management is getting out of the client, this lets you link to it

MSC4190 to make encrypted ASes work without /login

/login doesn't really work for application services and MAS, this fills the gap to work around that

MSC3824 for guidelines for existing clients

Filling some gaps to make sure you can provide a good enough experience

Features to support other existing use cases

Some features don't have to be an MSC.
Need an access token for a bot?
MAS should have a feature for that

What about the future?

MSC4108 signs you in and sets up E2EE with a QR code

Based on the OAuth 2.0 Device Code Grant
My boss loves showing off that demo!

Client credentials grant for application services

Replace the “hs_token” and “as_token” shared secrets with asymmetric keys

Widget as Matrix clients

We can now give restricted scopes, what if widgets were just regular Matrix clients?

Restricted API scopes

We now have the base to give restricted access to a subset of resource

Can I have it?

An overview of where we’re at, and a rough timeline

What
is ready today

- **MAS** support with **Synapse**
- Basic **migration tool**
- Comprehensive **MSCs**
- Solid **documentation**
- **Local password** accounts
- Upstream **OIDC SSO**
- **beta.matrix.org**
- Available in the
  Element Server Suite

What
is ready today

- **MAS** support with **Synapse**
- Basic **migration tool**
- Comprehensive **MSCs**
- Solid **documentation**
- **Local password** accounts
- Upstream **OIDC SSO**
- **beta.matrix.org**
- Available in the
  Element Server Suite

What
we are working on

- Get the **MSCs** to FCP
- Missing features for
  open homeservers
- Migrate **matrix.org**
- **Better migration** tools
- Features for **bots and
  automation** use cases

What is ready today

- **MAS** support with **Synapse**
- Basic **migration tool**
- Comprehensive **MSCs**
- Solid **documentation**
- **Local password** accounts
- Upstream **OIDC SSO**
- **beta.matrix.org**
- Available in the **Element Server Suite**

What we are working on

- Get the **MSCs** to FCP
- Missing features for **open homeservers**
- Migrate **matrix.org**
- **Better migration** tools
- Features for **bots and automation** use cases

What isn’t ready yet

- Community deployment tools
- Community server admin tools
- Encryption for Application Services
- Other homeserver implementations
- ~~Existing clients support~~ Next-gen auth capable clients

Get in touch!

MSC3861 **areweoidcyet.com** **#matrix-auth:matrix.org**

github.com/element-hq/matrix-authentication-service