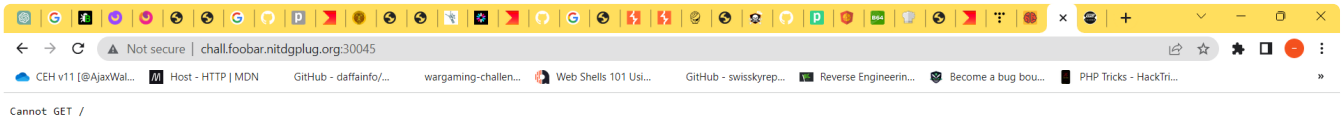
Foo-Bar CTF

Web-Inspect

Don't think, just push it to production....

When I opened this challenge for the first time, I thought the challenge was down

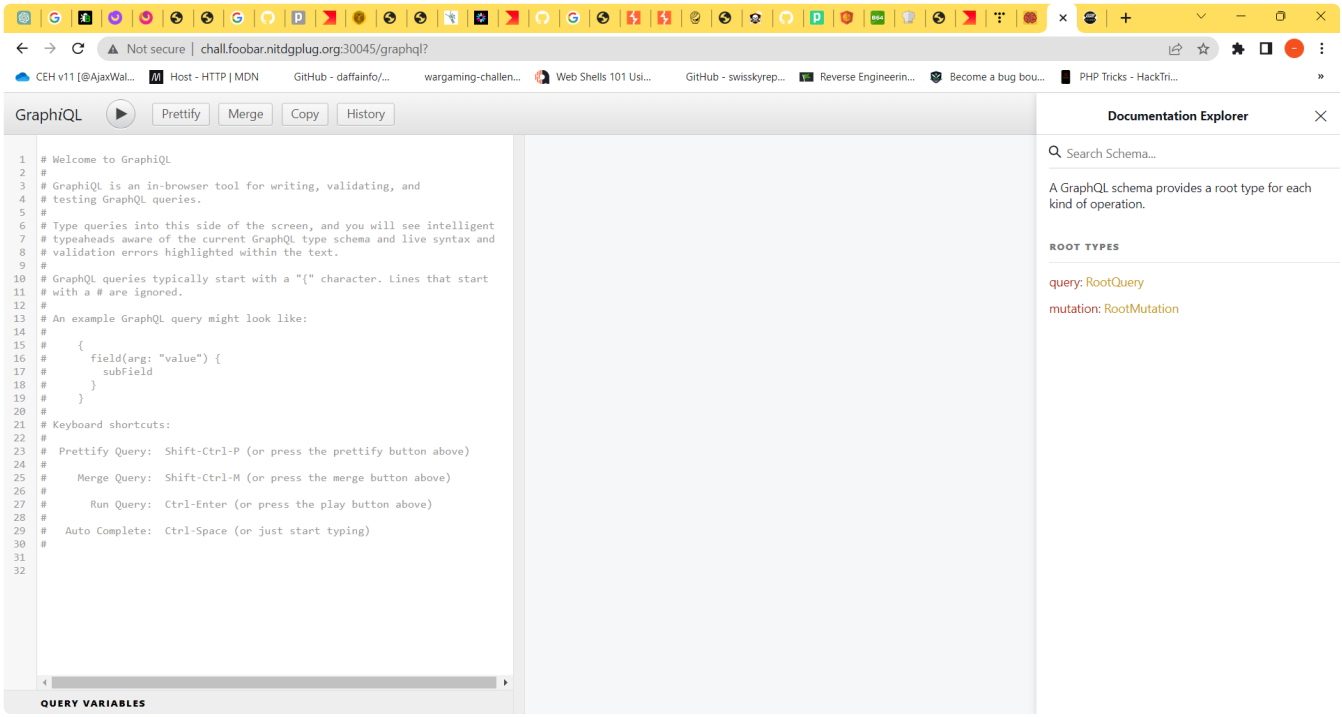


So I just left this one and started to try other challenges, but after a while, when I checked this challenge once again, some teams had solved it, but still the site looked the same. A new hint was also added to the challenge: "REST API IS BORING,SO I USED A MODERN ONE". From the hint, I figured that this challenge was something related to APIs. Basically An API is a set of defined rules that explain how computers or applications communicate with one another. API act as an intermediary between the web server and client and process data.

Now I have to find the correct API to access this site, so I started to search about REST API's alternatives, Modern API's,...

Luckily, I found a comparison between REST API vs. GraphQL in one of my Google search results.

So I quickly opened the following URL: chall.foobar.nitdgplug.org:30045/graphql.



Okay, now that I've confirmed that the site uses the GraphQL API, but where is the flag?

I was clueless here, but luckily I found a writeup related to this GraphQL: https://itsfading.github.io/posts/Hackerone-GraphQL-CTF-Writeup/

The first thing to do with this GraphQL API is to understand how it is structured. We need to provide an introspection query to get the structure and the available resources.

Introspection is the ability to query which resources are available in the current API schema.

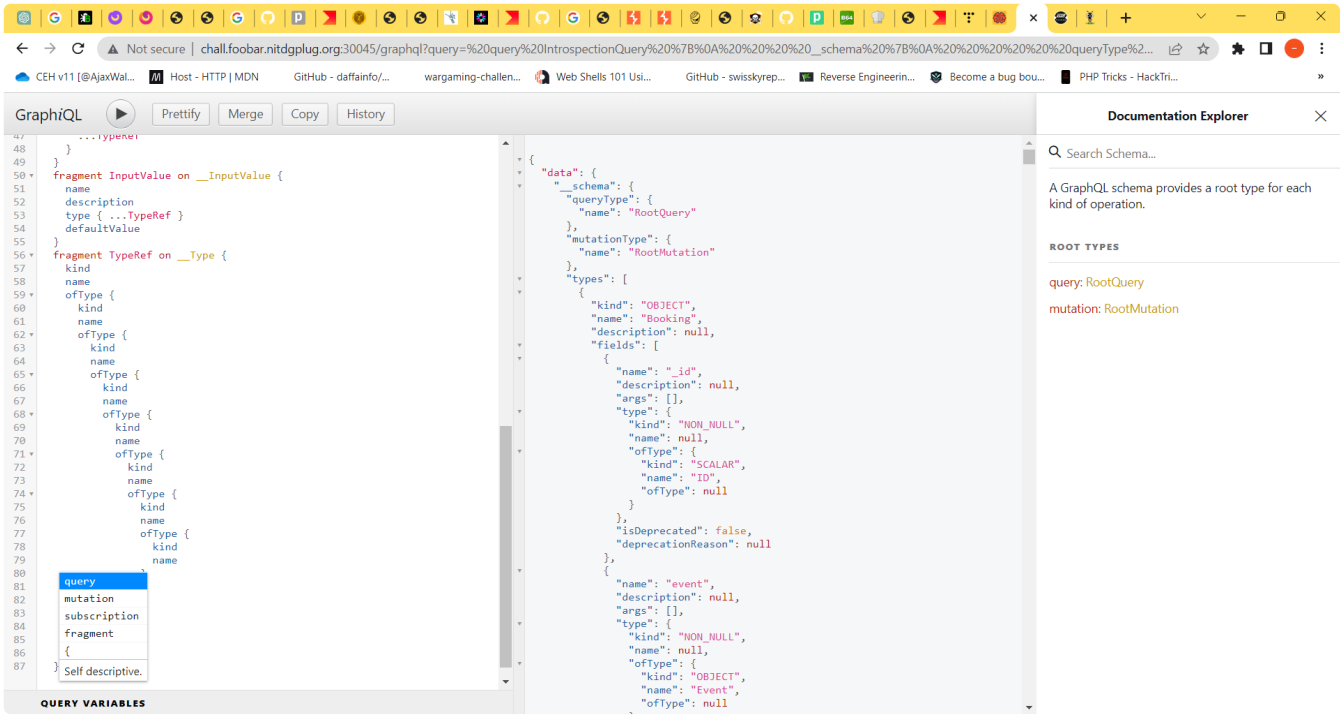
```
query IntrospectionQuery {
   __schema {
     queryType { name }
     mutationType { name }
     types {
       ...FullType
     }
     directives {
       name
       description
       locations
       args {
         ...InputValue
       }
     }
   }
 }
 fragment FullType on __Type {
   kind
   name
   description
   fields(includeDeprecated: true) {
     name

     description
     args {
       ...InputValue
     }
     type {
       ...TypeRef
     }
     isDeprecated
     deprecationReason
   }
   inputFields {
     ...InputValue
   }
   interfaces {
     ...TypeRef
   }
   enumValues(includeDeprecated: true) {
     name
     description
     isDeprecated
     deprecationReason
   }
   possibleTypes {
     ...TypeRef
   }
 }
 fragment InputValue on __InputValue {
   name
```

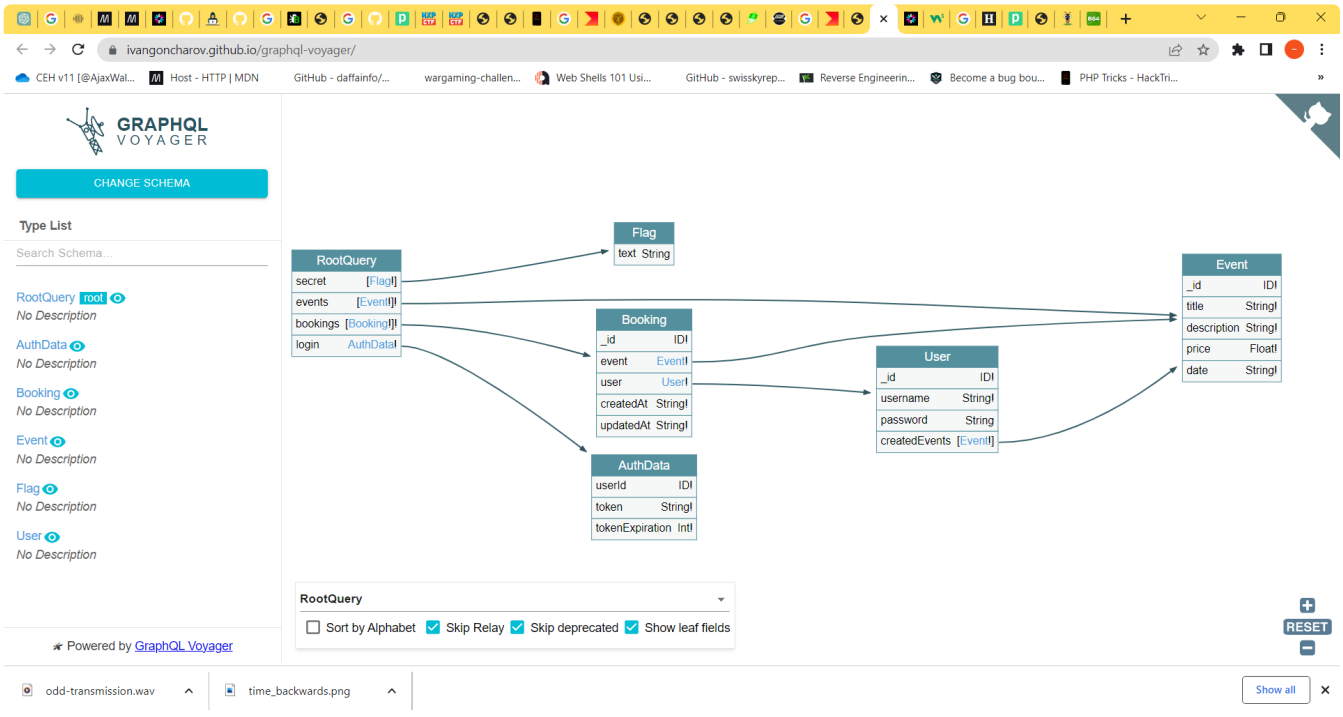
```
    description
    type { ...TypeRef }
    defaultValue
  }
  fragment TypeRef on __Type {
    kind
    name
    ofType {
      kind
      name
      ofType {
        kind
        name
        ofType {
          kind
          name
          ofType {
            kind
            name
            ofType {
              kind
              name
              ofType {
                kind
                name

                ofType {
                  kind
                  name
                }
              }
            }
          }
        }
      }
    }
  }
```

After passing this query, we get the rootquery, columns, and their datatype in the form of the query itself, like this:



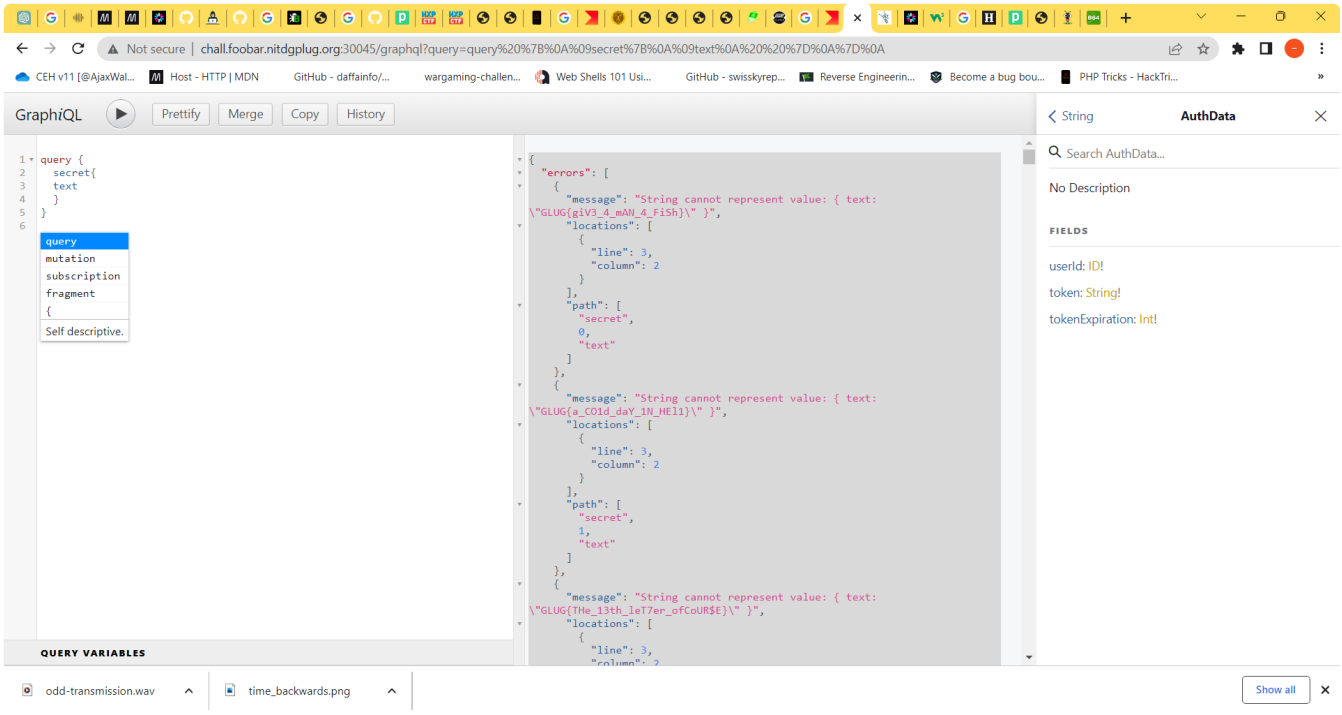
Difficult to read, right? Don't worry, we've got this amazing site https://apis.guru/graphql-voyager/ which transforms this output query into an interactive graph.



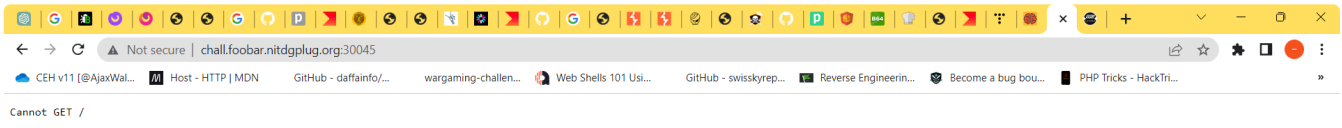
We got this beautiful graph upon pasting our output of introspection query, from this graph we can find the Flag column .

To print the Flag we need to write a simple GraphQL query like this

query { secret{ text } }



when I opened this challenge first time , I thought the challenge was down



So I just left this one and started to try other challenges but after a while when I checked this challenge once again some teams have solved this but still the site looked same.

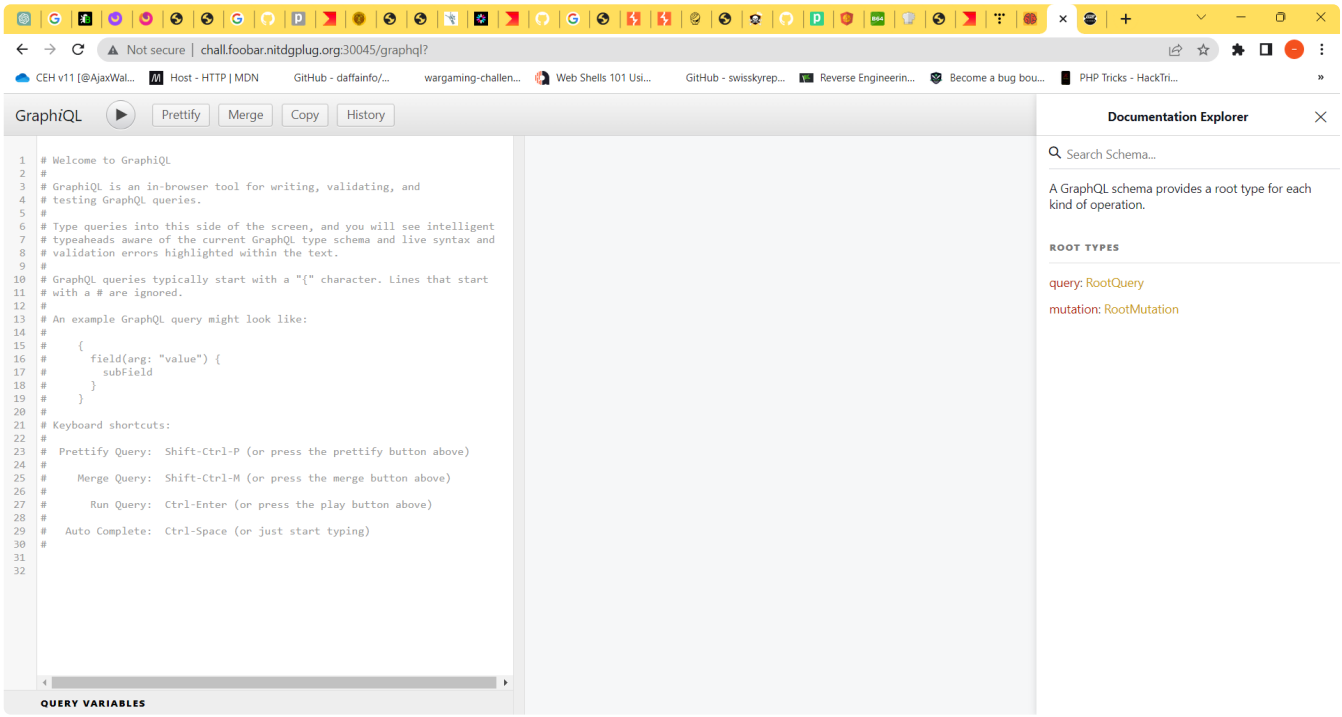
A new hint was also added to the challenge "REST API IS BORING , SO I USED A MODERN ONE".

From the hint I figured that this challenge is something related to API's. Basically An API is a set of defined rules that explain how computers or applications communicate with one another. API act as an intermediate between web server and client and process data.

Now I have to find the correct API to access this site, so started to search about REST API's alternatives, Modern API's ...

Luckily I founded a comparison between REST API vs GraphQL , in one of my google search results.

So quickly opened the following URL: chall.foobar.nitdgplug.org:30045/graphql



Now I confirmed that the site using GraphQL API ,okay but where is the flag?

I was clueless here but luckily I found a writeup related to this GraphQL
https://itsfading.github.io/posts/Hackerone-GraphQL-CTF-Writeup/

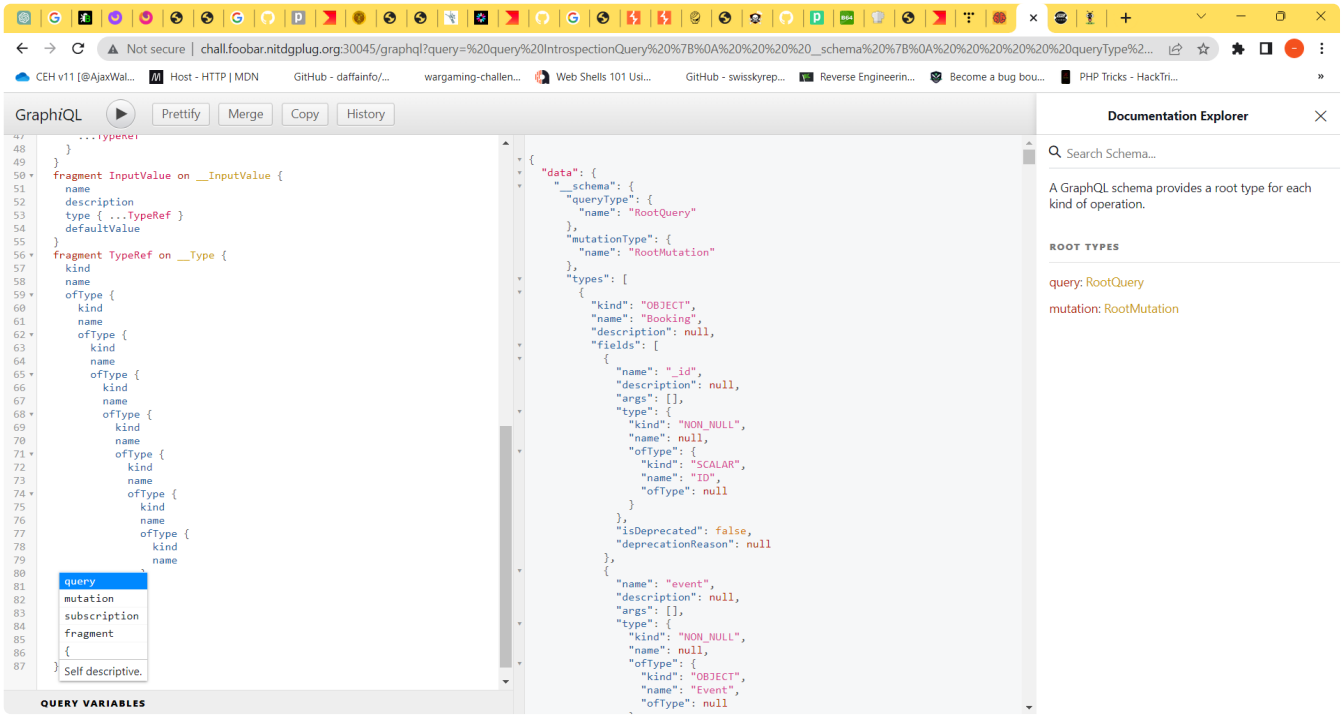
First step to do on this GraphQL API is to understand how it is structured , we need to provide introspection query to get the structure and the available resources.

Introspection is the ability to query which resources are available in the current API schema.

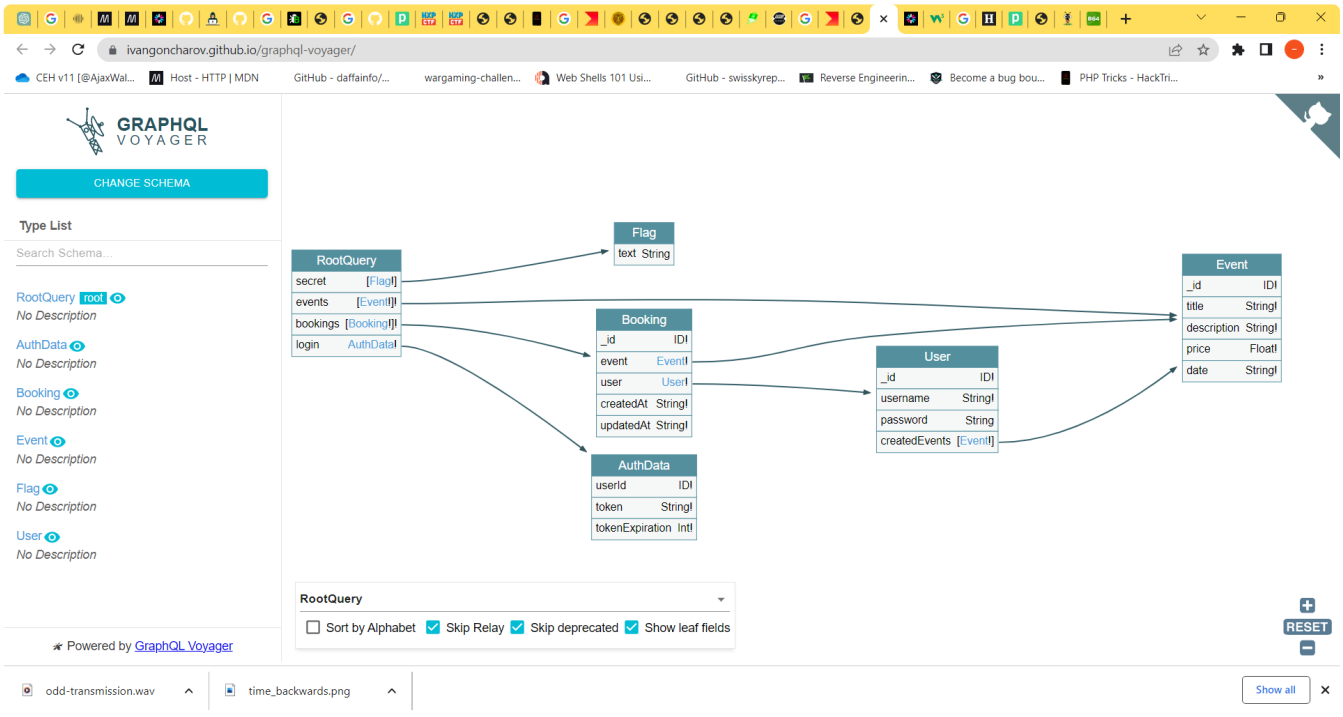
```
query IntrospectionQuery {
   __schema {
     queryType { name }
     mutationType { name }
     types {
       ...FullType
     }
     directives {
       name
       description
       locations
       args {
         ...InputValue
       }
     }
   }
 }
 fragment FullType on __Type {
   kind
   name
   description
   fields(includeDeprecated: true) {
     name
     description
     args {
       ...InputValue
     }
     type {
       ...TypeRef
     }
     isDeprecated
     deprecationReason
   }
   inputFields {
     ...InputValue
   }
   interfaces {
     ...TypeRef
   }
   enumValues(includeDeprecated: true) {
     name
     description
     isDeprecated
     deprecationReason
   }
   possibleTypes {
     ...TypeRef
   }
 }
 fragment InputValue on __InputValue {
   name
```

```
    description
    type { ...TypeRef }
    defaultValue
  }
  fragment TypeRef on __Type {
    kind
    name
    ofType {
      kind
      name
      ofType {
        kind
        name
        ofType {
          kind
          name
          ofType {
            kind
            name
            ofType {
              kind
              name
              ofType {
                kind
                name
                ofType {
                  kind
                  name
                }
              }
            }
          }
        }
      }
    }
  }
```

After passing this query , we get the rootquery ,coloumns and their datatype in the form of query itself like this,



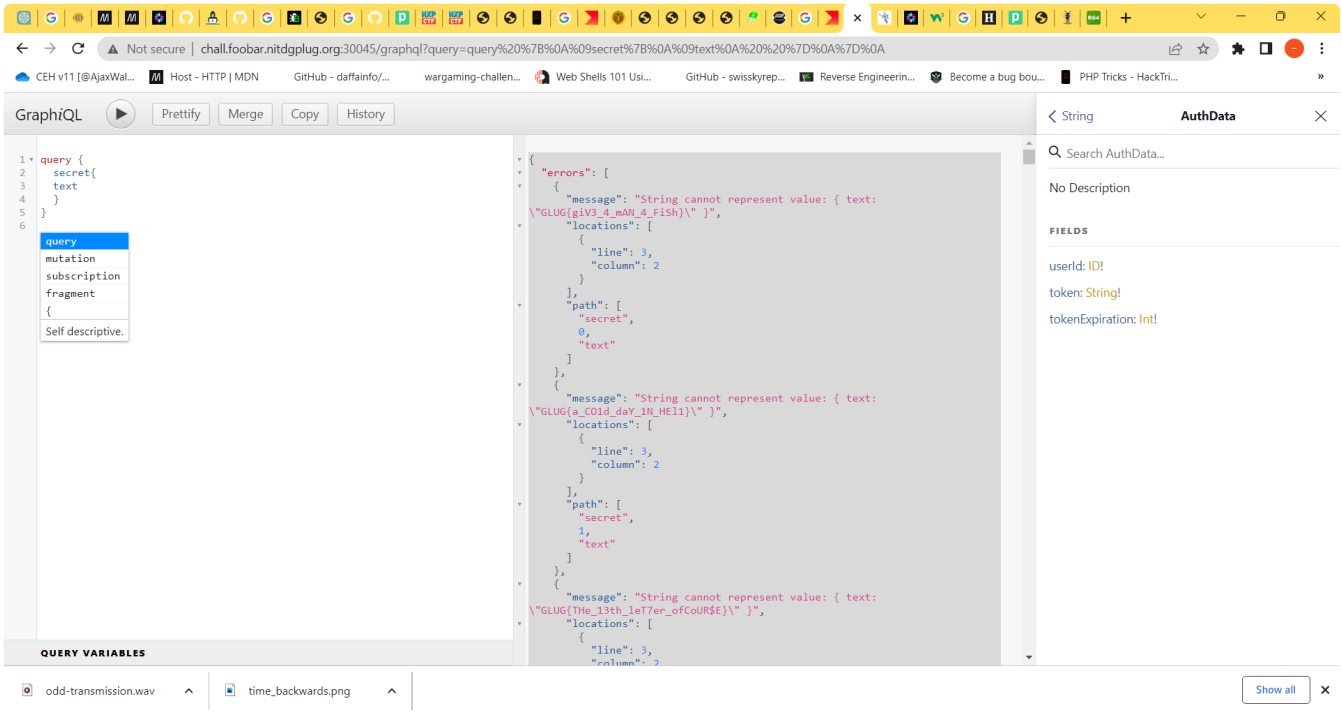
Difficult to read right?, don't worry we got this amazing site https://apis.guru/graphql-voyager/ which transforms this output query into an interactive graph .



We got this beautiful graph upon pasting our output of introspection query, from this graph we can find the Flag column .

To print the Flag we need to write a simple GraphQL query like this

query { secret{ text } }



I got like 74 flags which all of them are in correct format GLUG{}.So I wrote a python script sorted the flags and tried to submit the flags via intruder using burpsuite and Got the correct flag GLUG{1nsp3c7_1n_gr4phq6}.After the CTF , I learnt the intended solution , all flags are in text format but the actual flag is in flag format.