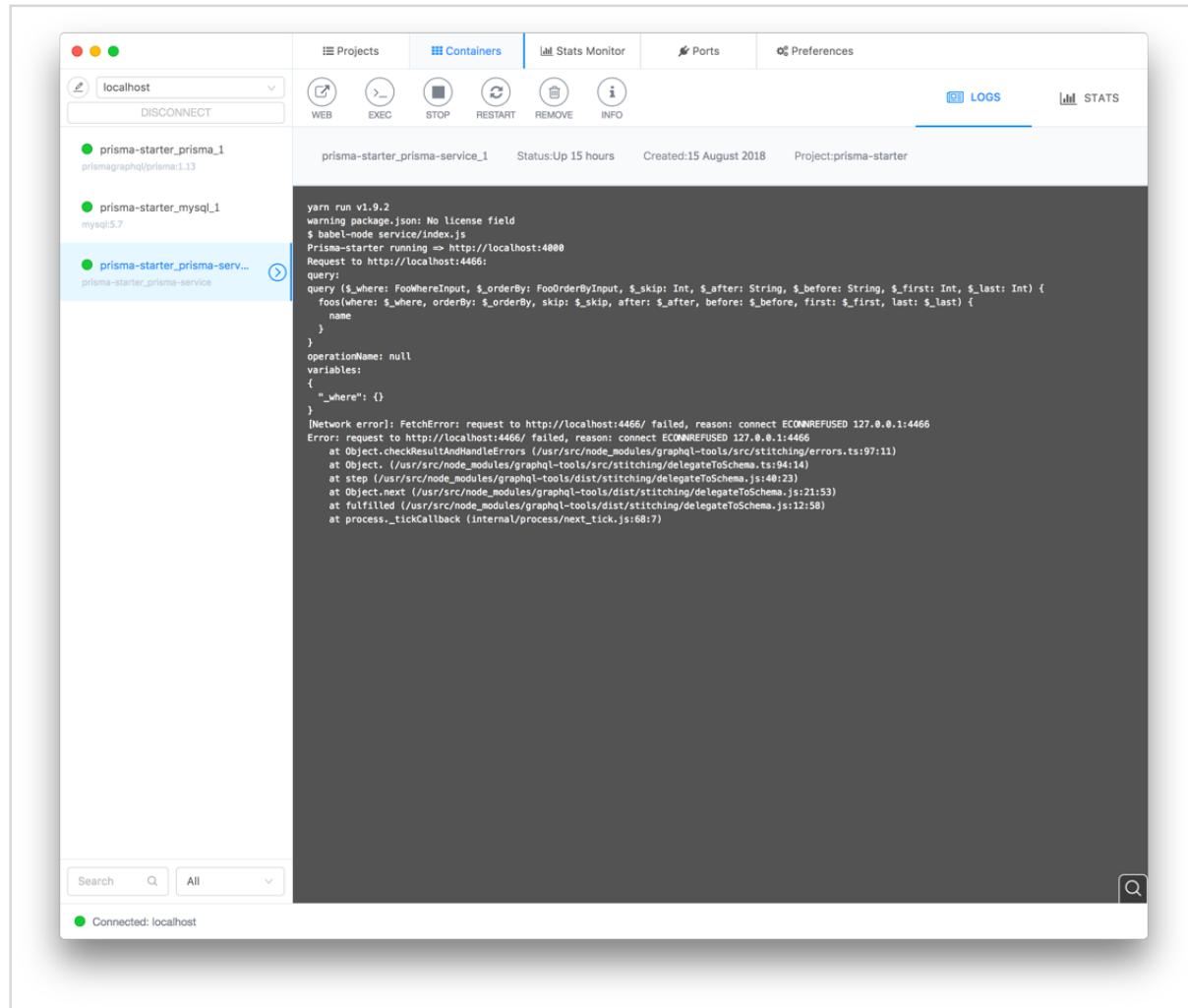


# Docker - Localhost Access Issue

#alechp/capclip

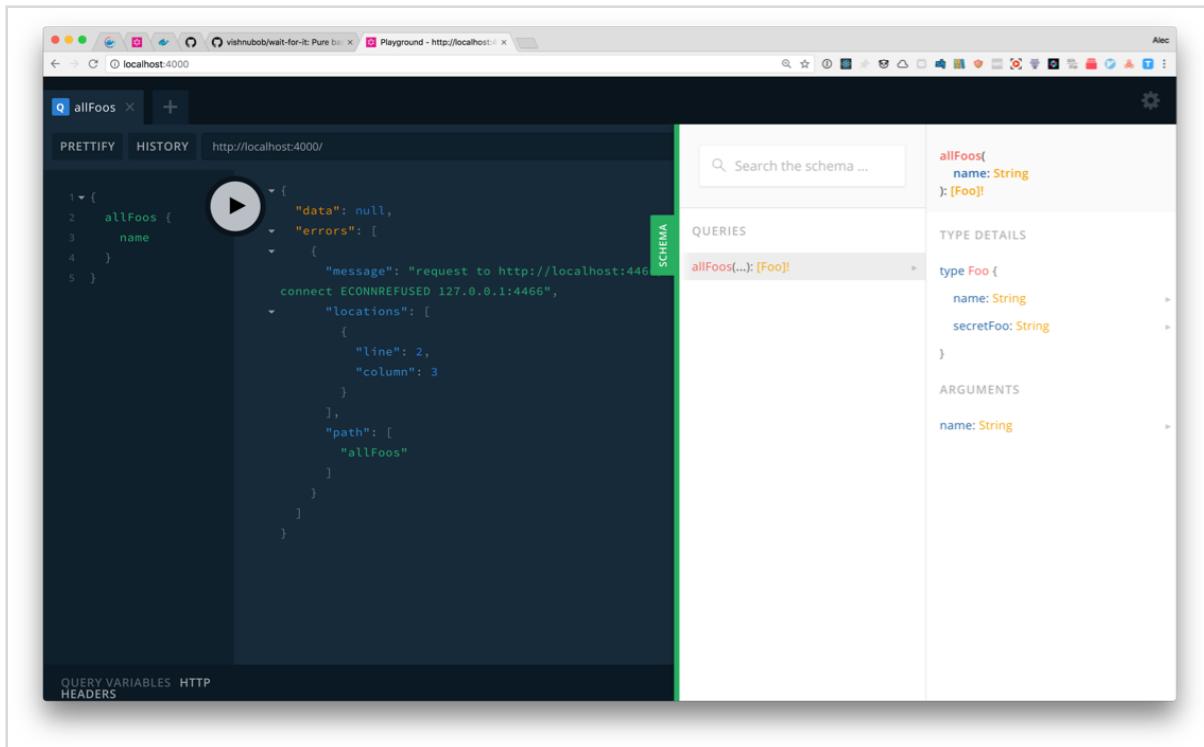
## Problem

localhost:4466 cannot be accessed from localhost:4000



## Attempt 1a

Prisma 1.8 Host '172.19.0.2' is not allowed to connect to this MySQL server on EC2 · Issue #2498 · prisma/prisma · GitHub



- Added environment. Still failed.

## Attempt 1b

- add command
  - install wait-for
- add restart

### Using `wait-for` with Docker-compose

```
command: >
  sh -c './wait-for prisma:4466 && prisma deploy && npm start'
```

### Using `wait-for-it` with Docker-compose

[Usage with Docker · Issue #57 · vishnubob/wait-for-it · GitHub](#)

```
#Dockerfile
COPY wait-for-it.sh /wait-for-it.sh
RUN chmod +x /wait-for-it.sh
```

```
my-service:
```

```
build:  
  context: "."  
  container_name: my_container  
  command: ["/wait-for-it.sh", "mysql:3306", "--", [entrypoint function]]
```

[GitHub - mal/docker-for-mac-host-bridge: Host-accessible containers with Docker for Mac](#)

This is a way to use host networking on mac. Home grown solution

More..

node.js - Prisma binding unable to connect to prisma server. request to <http://localhost:4466>/ failed, reason: connect ECONNREFUSED 127.0.0.1:4466 - Stack Overflow

Not sure if this can be done in isolation or whether everything else I tried below is necessary. `prisma-starter` has the code though

One thing I dislike is that I moved everything to root directory (everything previously in / database) so that I could run prisma deploy. Not sure this is a good thing. But figure it might be necessary since this is being distributed. Should be a way to better configure this in COPY of Dockerfile, but idk... low stakes problems for another day

**Close to done**

Ran into some issues with schema. Updated pathing.

The screenshot shows the GraphQL playground interface. At the top, there's a browser header with 'Playground - http://localhost:4000'. Below it is a search bar with 'localhost:4000'. The main area has tabs for 'PRETTIFY', 'HISTORY', and 'http://localhost:4000/'. There are buttons for 'COPY CURL' and 'SHARE PLAYGROUND'. A large play button icon is centered above the results. To the right, there's a 'SCHEMA' button. The results section shows a query 'allFoos' with a single result object:

```

1 ▾ {
2   allFoos(name:"foo"){
3     name
4   }
5 }
6
}
    "data": {
      "allFoos": [
        {
          "name": "Some foo"
        }
      ]
    }
  }
}

```

Below the results, there are sections for 'QUERY VARIABLES' and 'HTTP HEADERS'. At the bottom right, there's a 'TRACING' button.

The screenshot shows a terminal window with the following output:

```

1. alephp: bash -c "clear && docker exec -it prisma-starter_prisma-service_1 sh" (zsh)
# prisma deploy
Deploying service 'default' to stage 'default' to server 'default' 1.0s
Changes:
Post (Type)
- Deleted type 'Post'

User (Type)
- Deleted type 'User'

Foo (Type)
+ Created type 'Foo'
+ Created field 'id' of type 'GraphQLID!'
+ Created field 'name' of type 'String'
+ Created field 'secretFoo' of type 'String'
+ Created field 'updatedAt' of type 'Datetime!'
+ Created field 'createdAt' of type 'Datetime!'

PostToUser (Relation)
- Deleted relation between undefined and undefined

Applying changes 1.2s
Your Prisma GraphQL database endpoint is live:
HTTP: http://prisma:4466
WS: ws://prisma:4466

post-deploy:
warning command prepare both exists in plugin /usr/src/node_modules/graphql-cli-prepare and is shipped with the graphql-cli.
The plugin is being ignored.

Running graphql get-schema -p database ✓
warning command prepare both exists in plugin /usr/src/node_modules/graphql-cli-prepare and is shipped with the graphql-cli.
The plugin is being ignored.

Running graphql codegen...
".graphqlconfig" file is not available in the provided config directory: /usr/src
Please check the config directory.

Run graphql init to create new .graphqlconfig
Running graphql codegen ✘
# prisma seed
Seeding based on seed.graphql 369ms
#
```

The screenshot shows the Prisma Cloud interface with the 'LOGS' tab selected. The left sidebar lists projects and databases: 'localhost' (selected), 'prisma-starter\_prisma-service\_1', and 'prisma-starter\_mysql\_1'. The main area displays log entries for 'prisma-starter\_prisma-service\_1'. The log output is as follows:

```
[2]     "line": 8,
[2]     "column": 3
[2]
[2]
[2]
[2]
[2]
[2]   },
[2]   "status": 200
[2]
[2]
[2] Get in touch if you need help: https://www.prisma.io/forum/
[2] To get more detailed output run $ export DEBUG=*
[2] prisma seed exited with code 0
[3] Request to http://prisma:4466:
[3] query:
[3]   query { _where: FooWhereInput, _orderBy: FooOrderByInput, _skip: Int, _after: String, _before: String, _first: Int, _last: Int }
[3]   foos(_where: _where, _orderBy: _orderBy, skip: _skip, after: _after, before: _before, first: _first, last: _last) {
[3]     name
[3]   }
[3]
[3]   operationName: null
[3]   variables:
[3]   {
[3]     "_where": {
[3]       "name_contains": "foo"
[3]     }
[3]   }
[3] Response from http://prisma:4466:
[3] {
[3]   "foos": []
[3]
[3] Request to http://prisma:4466:
[3] query:
[3]   query { _where: FooWhereInput, _orderBy: FooOrderByInput, _skip: Int, _after: String, _before: String, _first: Int, _last: Int }
[3]   foos(_where: _where, _orderBy: _orderBy, skip: _skip, after: _after, before: _before, first: _first, last: _last) {
[3]     name
[3]   }
[3]
[3]   operationName: null
[3]   variables:
[3]   {
[3]     "_where": {
[3]       "name_contains": "foo"
[3]     }
[3]   }
[3] Response from http://prisma:4466:
[3] {
[3]   "foos": [
[3]     {
[3]       "name": "Some foo"
[3]     }
[3]   ]
[3]
```

Now the big problem is making sure prisma deploy and prisma seed are actually run in docker compose...

Works on rebuild...

I believe this works.

However, since I'm running prisms seed every time, it's adding duplicate records..

## first rebuild

A screenshot of the GraphQL playground interface. The URL in the address bar is `localhost:4000`. The query entered is:

```
1 ▼ {  
2   allFoos(name:"foo"){  
3     name  
4   }  
5 }  
6
```

The results pane shows the response:

```
1 ▼ {  
2   "data": {  
3     "allFoos": [  
4       {  
5         "name": "Some foo"  
6       },  
7       {  
8         "name": "Some foo"  
9       }  
10      ]  
11    }  
12  }  
13
```

Below the results, there are tabs for **QUERY VARIABLES**, **HTTP HEADERS**, and **TRACING**. A green **SCHEMA** button is visible on the right.

## second rebuild

A screenshot of the GraphQL playground interface. The URL in the address bar is `localhost:4000`. The query entered is:

```
1 ▼ {  
2   allFoos(name:"foo"){  
3     name  
4   }  
5 }  
6
```

The results pane shows the response:

```
1 ▼ {  
2   "data": {  
3     "allFoos": [  
4       {  
5         "name": "Some foo"  
6       },  
7       {  
8         "name": "Some foo"  
9       },  
10      ]  
11    }  
12  }  
13
```

Below the results, there are tabs for **QUERY VARIABLES**, **HTTP HEADERS**, and **TRACING**. A green **SCHEMA** button is visible on the right.

Notice how “some foo” record duplicated

Notice how :4466 doesn't show posts / data from previous db anymore. w00t :D

The screenshot shows the Prisma Studio interface running in a web browser. The URL is `localhost:4466`. The main area displays a GraphQL schema with the following code:

```
1 query {
2   foos {
3     name
4   }
5 }
6
7
8 # query {
9 #   __schema {
10 #     types {
11 #       fields {
12 #         name
13 #       }
14 #     }
15 #   }
16 }
```

The right side of the interface shows the results of a query. A play button icon is overlaid on the results. The results are:

```
1 {
2   "data": [
3     {
4       "foos": [
5         {
6           "name": "The homie"
7         },
8         {
9           "name": "The second homie"
10        },
11        {
12          "name": "Some foo"
13        },
14        {
15          "name": "The homie"
16        }
17      ]
18    }
19 }
```

Below the schema and results, there are tabs for "QUERY VARIABLES" and "HTTP HEADERS". On the right, there is a sidebar titled "SCHEMA" which contains a search bar and lists of queries, mutations, and connections.

## Results on destroyed/freshly built containers

### Error: Prisma Permission Access

Prisma doesn't have permissions

```

----> 18f0d6c1e1a8
Step 11/15 : RUN npm install -g prisma
--> Running in 957594738ea
/usr/local/bin/prisma -> /usr/local/lib/node_modules/prisma/dist/index.js

> prisma@1.14.0 postinstall /usr/local/lib/node_modules/prisma
> prisma init-prisma || echo ""

error displaying error
{ Error: EACCES: permission denied, mkdir '/root/.cache/prisma'
  at Object.mkdirSync (fs.js:729:3)
  at Object.mkdirsSync (/usr/local/lib/node_modules/prisma/node_modules/fs-extra/lib/mkdirs/mkdirs-sync.js:31:9)
  at dir (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Config.ts:245:6)
  at Config.get [as cacheDir] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Config.ts:102:15)
  at Output.get [as errlog] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Output/index.ts:158:34)
  at Output.LogError (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Output/index.ts:255:46)
  at Output.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Output/index.ts:174:12)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Output/index.js:48:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Output/index.js:21:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Output/index.js:15:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Output/index.js:11:12)
  at Output.error (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Output/index.js:213:16)
  at Object.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/CLI.ts:49:9)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/CLI.js:32:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/CLI.js:13:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/CLI.js:7:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/CLI.js:3:12)
  at process.handleErr (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/CLI.ts:45:30)
  at Object.onceWrapper (events.js:182:13)
  at process.emit (events.js:182:13)
  at process.emit (/usr/local/lib/node_modules/prisma/node_modules/source-map-support/source-map-support.js:439:21)
  at emitPromiseRejectionWarnings (internal/process/promises.js:105:20)
  at process._tickCallback (internal/process/next_tick.js:69:34)
  at Function.Module.runMain (internal/modules/cjs/loader.js:745:11)
  at startup (internal/bootstrap/node.js:266:19)
  at bootstrapNodeJSCore (internal/bootstrap/node.js:596:3)

errno: -13,
syscall: 'mkdir',
code: 'EACCES',
path: '/root/.cache/prisma' }

Error: EACCES: permission denied, mkdir '/root/.cache/prisma'
  at Object.mkdirSync (fs.js:729:3)
  at Object.mkdirsSync (/usr/local/lib/node_modules/prisma/node_modules/fs-extra/lib/mkdirs/mkdirs-sync.js:31:9)
  at dir (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Config.ts:245:6)
  at Config.get [as cacheDir] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Config.ts:102:15)
  at Lock.get [as updateLockfile] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Plugin/Lock.ts:16:64)
  at Lock.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Plugin/Lock.ts:24:28)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:32:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:13:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:7:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:3:12)
  at Lock.unread (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:60:16)
  at Lock.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Plugin/Lock.ts:35:16)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:32:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:13:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:7:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:3:12)
  at Lock.upgrade (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Lock.js:86:16)
  at Cache.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Plugin/Cache.ts:176:36)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Cache.js:32:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Cache.js:13:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Cache.js:7:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Cache.js:3:12)
  at Cache.fetchManagers (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Cache.js:175:16)
  at Plugins.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Plugin/Plugins.ts:42:37)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Plugins.js:32:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Plugins.js:13:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Plugins.js:7:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Plugins.js:3:12)
  at Plugins.load (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Plugin/Plugins.js:56:16)
  at PluginCommandManager.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/src/Dispatcher/Dispatcher.ts:66:19)
  at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Dispatcher/Dispatcher.js:42:23)
  at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Dispatcher/Dispatcher.js:23:53)
  at /usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Dispatcher/Dispatcher.js:17:71
  at new Promise (<anonymous>)
  at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Dispatcher/Dispatcher.js:13:12)
  at PluginCommandManager.findCommand (/usr/local/lib/node_modules/prisma/node_modules/prisma-clи-engine/dist/Dispatcher/Dispatcher.js:153:16)

```

~/Code/servexyz/starter/repositories/prisma-starter

master

```

at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:3:12)
at Lock.unread (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:60:16)
at Lock.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Plugin/lock.ts:35:16)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:32:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:13:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:7:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:3:12)
at Lock.upgrade (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/lock.js:86:16)
at Cache.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Plugin/cache.ts:176:36)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/cache.js:32:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/cache.js:13:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/cache.js:7:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/cache.js:3:12)
at Cache.fetchManagers (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Plugin/plugins.ts:175:16)
at Plugins.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Plugin/plugins.ts:42:37)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/plugins.js:32:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/plugins.js:13:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/plugins.js:7:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/plugins.js:3:12)
at Plugins.load (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/plugins.js:56:16)
at PluginCommandManager.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Dispatcher/Dispatcher.ts:66:19)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:42:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:23:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:17:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:13:12)
at PluginCommandManager.findCommand (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:153:16)
at Dispatcher.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Dispatcher/Dispatcher.ts:99:35)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:42:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:23:53)
at fulfilled (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:14:58)
at process._tickCallback (internal/process/next_tick.js:68:7)
at Function.Module.runMain (internal/modules/cjs/loader.js:745:11)
at startup (internal/bootstrap/node.js:266:19)
at bootstrapNodeJSCore (internal/bootstrap/node.js:596:3)

errno: -13,
syscall: 'mkdir',
code: 'EACCES',
path: '/root/.cache/prisma'

(node:25) UnhandledPromiseRejectionWarning: Error: EACCES: permission denied, mkdir '/root/.cache/prisma'
at Object.mkdirSync (fs.js:729:3)
at Object.mkdirsSync (/usr/local/lib/node_modules/prisma/node_modules/fs-extra/lib/mkdirs/mkdirs-sync.js:31:9)
at dir (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:245:6)
at Config.get [as cacheDir] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:102:15)
at Config.get [as requestsCachePath] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:115:27)
at StatusChecker.checkStatus (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/StatusChecker.ts:87:38)
at Output.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Output/index.ts:203:21)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/output/index.js:48:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/output/index.js:21:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/output/index.js:15:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/output/index.js:11:12)
at Output.error (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/output/index.js:213:16)
at Object.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/CLI.ts:49:9)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:32:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:13:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:7:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:3:12)
at process._handleErr (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/CLI.ts:45:30)
at Object.onceWrapper (events.js:173:13)
at process.emit [as _emit] (events.js:182:13)
at process.emit (/usr/local/lib/node_modules/prisma/node_modules/source-map-support/source-map-support.js:439:21)
at emitPromiseRejectionWarnings (internal/process/promises.js:105:20)
at process._tickCallback (internal/process/next_tick.js:69:34)
at Function.Module.runMain (internal/modules/cjs/loader.js:745:11)
at startup (internal/bootstrap/node.js:266:19)
at bootstrapNodeJSCore (internal/bootstrap/node.js:596:3)

(node:25) UnhandledPromiseRejectionWarning: Unhandled promise rejection. This error originated either by throwing inside of an async function without a catch block, or by rejecting a promise which was not handled with .catch(). (rejection id: 3)
(node:25) [DEP0018] DeprecationWarning: Unhandled promise rejections are deprecated. In the future, promise rejections that are not handled will terminate the Node.js process with a non-zero exit code.

```

prisma update check failed  
 Try running with sudo or get access  
 to the local update config store via  
 sudo chown -R \$USER:\$id -gn \$USER /root/.config

~/Code/servexyz/starter/.repositories/prisma-starter

master

```

at Plugins.load (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Plugin/Plugins.js:56:16)
at PluginCommandManager.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Dispatcher/Dispatcher.ts:66:19)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:42:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:23:53)
at at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:13:12)
at PluginCommandManager.findCommand (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:153:16)
at Dispatcher.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Dispatcher/Dispatcher.ts:99:35)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:42:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:23:53)
at fulfilled (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Dispatcher/Dispatcher.js:14:58)
at process._tickCallback (internal/process/next_tick.js:68:7)
at Function.Module.runMain (internal/modules/cjs/loader.js:745:11)
at startup (internal/bootstrap/node.js:266:19)
at bootstrapNodeJSCore (internal/bootstrap/node.js:596:3)
errno: -13
 syscall: 'mkdir',
 code: 'EACCES',
 path: '/root/.cache/prisma' )
(node:25) UnhandledPromiseRejectionWarning: Error: EACCES: permission denied, mkdir '/root/.cache/prisma'
at Object.mkdirSync (fs.js:729:3)
at Object.mkdirsSync (/usr/local/lib/node_modules/prisma/node_modules/fs-extra/lib/mkdirs/mkdirs-sync.js:31:9)
at dir (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:245:6)
at Config.get [as cacheDir] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:102:15)
at Config.get [as requestsCachePath] (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Config.ts:115:27)
at StatusChecker.checkStatus (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/StatusChecker.ts:87:38)
at Output.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/Output/index.ts:203:21)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Output/index.js:48:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Output/index.js:21:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Output/index.js:15:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Output/index.js:11:12)
at Output.error (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/Output/index.js:213:16)
at Object.<anonymous> (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/CLI.ts:49:9)
at step (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:32:23)
at Object.next (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:13:53)
at /usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:7:71
at new Promise (<anonymous>)
at __awaiter (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/dist/CLI.js:3:12)
at process.handleErr (/usr/local/lib/node_modules/prisma/node_modules/prisma-clie-engine/src/CLI.ts:45:30)
at Object.onceWrapper (events.js:273:13)
at process.emit (events.js:182:13)
at process.emit (/usr/local/lib/node_modules/prisma/node_modules/source-map-support/source-map-support.js:439:21)
at emitPromiseRejectionWarnings (internal/process/promises.js:105:28)
at process._tickCallback (internal/process/next_tick.js:69:34)
at Function.Module.runMain (internal/modules/cjs/loader.js:745:11)
at startup (internal/bootstrap/node.js:266:19)
at bootstrapNodeJSCore (internal/bootstrap/node.js:596:3)
(node:25) UnhandledPromiseRejectionWarning: Unhandled promise rejection. This error originated either by throwing inside of an async function without a catch block, or by rejecting a promise which was not handled with .catch(). (rejection id: 3)
(node:25) [DEP0018] DeprecationWarning: Unhandled promise rejections are deprecated. In the future, promise rejections that are not handled will terminate the Node.js process with a non-zero exit code.

prisma update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$ID -gN $USER /root/.config

npm WARN ajv-keywords@3.2.0 requires a peer of ajv@6.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.4 (node_modules/prisma/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.4: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
+ prisma@1.14.0
added 526 packages from 308 contributors in 17.67s
Removing intermediate container 957594738ea
--> 8535cff14193
Step 12/15 : RUN chmod +x ./wait-for-it.sh
--> Running in f5012cb085b5
Removing intermediate container f5012cb085b5
--> 8c17c13520b6
Step 13/15 : RUN yarn install
--> Running in 90e793872285
yarn install v1.9.2
warning package.json: No license field
warning prisma-starter: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
yarn install done in 0s

  ➜ ~/Code/servexyz/starter/.repositories/prisma-starter
  ✓ master

```

## Solution: Prisma Root Error

### Workaround 1

npm install failing (Error: EACCES: permission denied, mkdir) · Issue #21 · gdotdesign/elm-github-install · GitHub

if you install a package that you have the package name, you run 'sudo npm install package\_name --unsafe-perm=true --allow-root'. If you don't know which package you missing, just run 'sudo npm install --unsafe-perm=true --allow-root' in your project folder that has node\_modules.

## Workaround 2

EACCES: permission denied, open '/home/username/.cache/prisma/error.log' · Issue #1908 · prisma/prisma · GitHub

## Workaround 3 (What I chose)

use `yarn` to add globally instead of `npm`

```
~/Code/servxyz/starter/prisma-starter master*
> drebuid
Building prisma-service
Step 1/15 : FROM node:10
--> b064644cf368
Step 2/15 : WORKDIR /usr/src
--> Using cache
--> 12bb8a62863c
Step 3/15 : COPY src/package.json .
--> Using cache
--> 5573c2a00dc3
Step 4/15 : COPY src/yarn.lock .
--> Using cache
--> 54e22af0e2b
Step 5/15 : COPY prisma.yml .
--> Using cache
--> 12bf95e4be10
Step 6/15 : COPY seed.graphql .
--> Using cache
--> fd4f0d48d512
Step 7/15 : COPY datamodel.graphql .
--> Using cache
--> 5ed2282de678
Step 8/15 : COPY .env .
--> Using cache
--> 3188dc29c49d
Step 9/15 : COPY wait-for-it.sh .
--> Using cache
--> 0f03e13c44bd
Step 10/15 : RUN yarn global add concurrently
--> Running in 770a17d32ada
yarn global v1.9.2
warning package.json: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
success Installed "concurrently@3.6.1" with binaries:
  - concurrent
  - concurrently
Done in 2.41s.
Removing intermediate container 770a17d32ada
--> 6f2241d606ea
Step 11/15 : RUN yarn global add prisma
--> Running in 3fb6be8aa77f
yarn global v1.9.2
warning package.json: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
info fsevents@1.2.4: The platform "linux" is incompatible with this module.
info "fsevents@1.2.4" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
warning "prisma > prisma-clc-core@1.0.71" has unmet peer dependency "graphql@^0.12.0 || ^0.13.0".
[4/4] Building fresh packages...
success Installed "prisma@1.14.2" with binaries:
  - prisma
Done in 12.50s.
Removing intermediate container 3fb6be8aa77f
--> e9bcc7a8eae7
Step 12/15 : RUN chmod +x ./wait-for-it.sh
--> Running in 2ef85b7eb453
Removing intermediate container 2ef85b7eb453
--> 911e7cf76ab7
Step 13/15 : RUN yarn install
--> Running in 5a35416d5fc6
yarn install v1.9.2
warning package.json: No license field
warning prisma-starter: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
info fsevents@1.2.4: The platform "linux" is incompatible with this module.
info "fsevents@1.2.4" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
warning prisma-binding > graphql-import@0.4.5" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
info "prisma-binding > prisma-binding@1.5.19" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
warning "prisma-binding > apollo-link@1.2.1" has unmet peer dependency "graphql@^0.11.3 || ^0.12.3 || ^0.13.0".
warning "prisma-binding > graphql-binding@1.2.5" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".

```

~/Code/servxyz/starter/.repositories/prisma-starter

master 2

```

----> Using cache
--> 5ad2202d678
Step 8/15 : COPY .env .
----> Using cache
--> 3188dc29c49d
Step 9/15 : COPY wait-for-it.sh .
----> Using cache
--> 0f03e13c44bd
Step 10/15 : RUN yarn global add concurrently
----> Running in 770a17d32ada
yarn global v1.9.2
warning package.json: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
success Installed "concurrently@3.6.1" with binaries:
  - concurrent
  - concurrently
Done in 2.41s.
Removing intermediate container 770a17d32ada
--> 6f2241d606ea
Step 11/15 : RUN yarn global add prisma
----> Running in 3fb6be8aa77f
yarn global v1.9.2
warning package.json: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
info "fsevents@1.2.4: The platform "linux" is incompatible with this module.
info "fsevents@1.2.4" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
warning "prisma > prisma-clis-core@1.0.71" has unmet peer dependency "graphql@^0.12.0 || ^0.13.0".
[4/4] Building fresh packages...
success Installed "prisma@1.14.2" with binaries:
  - prisma
Done in 12.50s.
Removing intermediate container 3fb6be8aa77f
--> e8bcc7a8ea7
Step 12/15 : RUN chmod +x ./wait-for-it.sh
----> Running in 2ef85b7eb453
Removing intermediate container 2ef85b7eb453
--> 911e7cff6ab7
Step 13/15 : RUN yarn install
----> Running in 5a35416d5fc6
yarn install v1.9.2
warning package.json: No license field
warning prisma-starter: No license field
[1/4] Resolving packages...
[2/4] Fetching packages...
(node:8) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.
info "fsevents@1.2.4: The platform "linux" is incompatible with this module.
info "fsevents@1.2.4" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
warning "prisma-binding > graphql-import@0.4.5" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
warning " " > prisma-binding@1.5.19" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
warning "prisma-binding > apollo-link@1.2.1" has unmet peer dependency "graphql@^0.11.3 || ^0.12.3 || ^0.13.0".
warning "prisma-binding > graphql-binding@1.2.5" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
warning "prisma-binding > graphql-tools@2.21.0" has unmet peer dependency "graphql@^0.11.0 || ^0.12.0 || ^0.13.0".
warning "prisma > prisma-clis-core@1.0.42" has unmet peer dependency "graphql@^0.12.0 || ^0.13.0".
[4/4] Building fresh packages...
success Saved lockfile.
Done in 14.69s.
Removing intermediate container 5a35416d5fc6
--> 7e8f7cc50cdf
Step 14/15 : COPY src/service/ ./service/
--> 4a388bd1d651
Step 15/15 : EXPOSE 4000
----> Running in b544f57df7b4
Removing intermediate container b544f57df7b4
--> 3cb9ed456331
Successfully built 3cb9ed456331
Successfully tagged prisma-starter_prisma-service:latest
Recreating prisma-starter_mysql_1 ... done
Recreating prisma-starter_prisma_1 ... done
Recreating prisma-starter_prisma-service_1 ... done
~/Code/serveyx/starter/prisma-starter master* 1m 14s
> |
```

~/Code/serveyx/starter/repositories/prisma-starter

V master 2

Seems like it's still persisting (notice the duplicate data again):

The screenshot shows the GraphQL Playground interface running at `http://localhost:4000`. The query entered is:

```
1 ▼ {  
2   allFoos(name:"the"){  
3     name  
4   }  
5 }  
6 |
```

The response is a JSON object with the following structure:

```
▼ {  
  "data": {  
    "allFoos": [  
      {  
        "name": "The homie"  
      },  
      {  
        "name": "The second homie"  
      },  
      {  
        "name": "The homie"  
      },  
      {  
        "name": "The second homie"  
      },  
      {  
        "name": "The homie"  
      },  
      {  
        "name": "The second homie"  
      },  
      {  
        "name": "The homie"  
      },  
      {  
        "name": "The second homie"  
      }  
    ]  
  }  
}
```

The interface includes tabs for PRETTIFY, HISTORY, COPY CURL, SHARE PLAYGROUND, and SCHEMA. At the bottom, there are buttons for QUERY VARIABLES, HTTP HEADERS, and TRACING.

After, getting rid of local MySQL and reinstalling, now the error is back to its old ways...

The screenshot shows the GraphQL Playground interface running in a browser. The URL is `http://localhost:4000`. The query entered is:

```
1 ▼ {  
2   allFoos(name:"the"){  
3     name  
4   }  
5 }  
6
```

The response is displayed as a JSON object with a play button icon:

```
{  
  "data": null,  
  "errors": [  
    {  
      "message": "Project not found:  
'default@default'",  
      "locations": [],  
      "path": [  
        "foos"  
      ],  
      "code": 3016,  
      "requestId":  
        "local:api:cjl2un8hd00010b38mv3taocc"  
    }  
  ]  
}
```

Below the code editor, there are tabs for **QUERY VARIABLES**, **HTTP HEADERS**, and **TRACING**. On the right side, there is a vertical sidebar with a gear icon and a green tab labeled **SCHEMA**.

The screenshot shows a GraphQL playground interface running at <http://localhost:4466>. The left sidebar contains a search bar with the query 'foos' and a '+' button. Below it are tabs for 'PRETTYFY' and 'HISTORY'. The main area displays a GraphQL query:

```
query {
  foos {
    name
  }
}

# query {
#   __schema {
#     types {
#       fields {
#         name
#       }
#     }
#   }
}
```

To the right of the query is a large play button icon. To its right, the response is shown as a JSON object:

```
{
  "data": {
    "foos": [
      {
        "name": "The homie"
      },
      {
        "name": "The second homie"
      },
      {
        "name": "Some foo"
      },
      {
        "name": "The homie"
      }
    ]
  }
}
```

A green vertical bar labeled 'SCHEMA' is positioned next to the first 'name' field in the JSON response. At the bottom of the interface, there are buttons for 'QUERY VARIABLES' and 'HTTP HEADERS'.

- 4000 isn't recognizing project / saying project not found
    - 4466 isn't loading schema – meaning prisma deploy isn't running

```
command: './wait-for-it.sh prisma:4466 -- concurrently "ls -a /usr/src" "ls -a /usr/src/service" "prisma seed" "yarn start"'
```

^- Noticed that prisma deploy wasn't included.

Added:

```
command: './wait-for-it.sh prisma:4466 -- concurrently "ls -a /usr/src" "ls -a /usr/src/service" "prisma deploy" "prisma seed" "yarn start"'
```

## Now it works

The screenshot shows the Prisma GraphQL playground interface at `localhost:4000`. A query is being executed:

```
1 ▾ {  
2   allFoos(name:"the"){  
3     name  
4   }  
5 }  
6
```

The results are displayed in a tree view:

```
1 {  
2   "data": {  
3     "allFoos": [  
4       {  
5         "name": "The homie"  
6       },  
7       {  
8         "name": "The second homie"  
9       }  
10      ]  
11    }  
12  }
```

Below the results, there are buttons for "QUERY VARIABLES" and "HTTP HEADERS". On the right side, there's a "SCHEMA" button and a "TRACING" button.

The screenshot shows the Prisma GraphQL playground interface at `localhost:4466`. A more complex query is being executed:

```
1 ▾ query {  
2   foos {  
3     name  
4   }  
5 }  
6  
7 # query {  
8 #   __schema {  
9 #     types {  
10 #       fields {  
11 #         name  
12 #       }  
13 #     }  
14 #   }  
15 # }
```

The results are displayed in a tree view:

```
1 {  
2   "data": {  
3     "foos": [  
4       {  
5         "name": "The homie"  
6       },  
7       {  
8         "name": "The second homie"  
9       },  
10      {  
11        "name": "Some foo"  
12      }  
13    ]  
14  }  
15 }
```

On the right side, there is a search bar labeled "Search the schema ...", a sidebar with sections for "QUERIES" and "MUTATIONS" containing various mutation names, and buttons for "QUERY VARIABLES" and "HTTP HEADERS".

Question: will this continue prisma seed new data every time...?

The screenshot shows the Prisma GraphQL playground interface. The URL in the browser is `localhost:4000`. The query entered is:

```
1 ▼ {
2   allFoos(name:"the"){
3     name
4   }
5 }
```

The results pane displays the JSON response from the API:

```
{
  "data": {
    "allFoos": [
      {
        "name": "The homie"
      },
      {
        "name": "The second homie"
      },
      {
        "name": "The homie"
      },
      {
        "name": "The second homie"
      },
      {
        "name": "The homie"
      },
      {
        "name": "The second homie"
      }
    ]
  }
}
```

Below the results, there are tabs for **QUERY VARIABLES**, **HTTP HEADERS**, and **TRACING**. A vertical sidebar on the right is labeled **SCHEMA**.

The screenshot shows the Prisma Playground interface running at <http://localhost:4466/>. The main area displays a GraphQL query for a "foos" type:

```
query {
  foos {
    name
  }
}

# query {
#   __schema {
#     types {
#       fields {
#         name
#       }
#     }
#   }
# }
```

The results of the query are shown in a JSON-like structure under the "data" key:

```
"data": {  
  "foos": [  
    {  
      "name": "The homie"  
    },  
    {  
      "name": "The second homie"  
    },  
    {  
      "name": "Some foo"  
    },  
    {  
      "name": "The homie"  
    },  
    {  
      "name": "The second homie"  
    },  
    {  
      "name": "Some foo"  
    },  
    {  
      "name": "The homie"  
    },  
    {  
      "name": "The second homie"  
    },  
    {  
      "name": "Some foo"  
    }  
  ]  
}
```

A large play button icon is overlaid on the results. To the right of the results, there is a sidebar titled "SCHEMA" which contains a search bar and a list of schema components:

- QUERIES
  - foos(...): [Foo]!
  - foo(...): Foo
  - foosConnection(...): FooConnection!
  - node(...): Node
- MUTATIONS
  - createFoo(...): Foo!
  - updateFoo(...): Foo
  - deleteFoo(...): Foo!
  - upsertFoo(...): Foo!
  - updateManyFoos(...): BatchPayload!
  - deleteManyFoos(...): BatchPayload!
- SUBSCRIPTIONS
  - foo(...): FooSubscriptionPayload

At the bottom left, there are buttons for "QUERY VARIABLES" and "HTTP HEADERS".