**Body of Knowledge (BOK)**

Mohammed Nasser Alshukaili.

ICT, Fontys University of Applied Sciences.

Cyber Security, Semester 4.

September 2022

Contents

[INTRODUCTION 3](#_Toc113407010)

[IT Background 3](#_Toc113407011)

[(BOK) Web Application Security (week 1-4) 4](#_Toc113407012)

[Path Traversal and Command Injection 5](#_Toc113407013)

[Brute Force 5](#_Toc113407014)

[(BOK) Network Security (week 5-8) 8](#_Toc113407015)

[(BOK) Security Concepts (week 9-10) 8](#_Toc113407016)

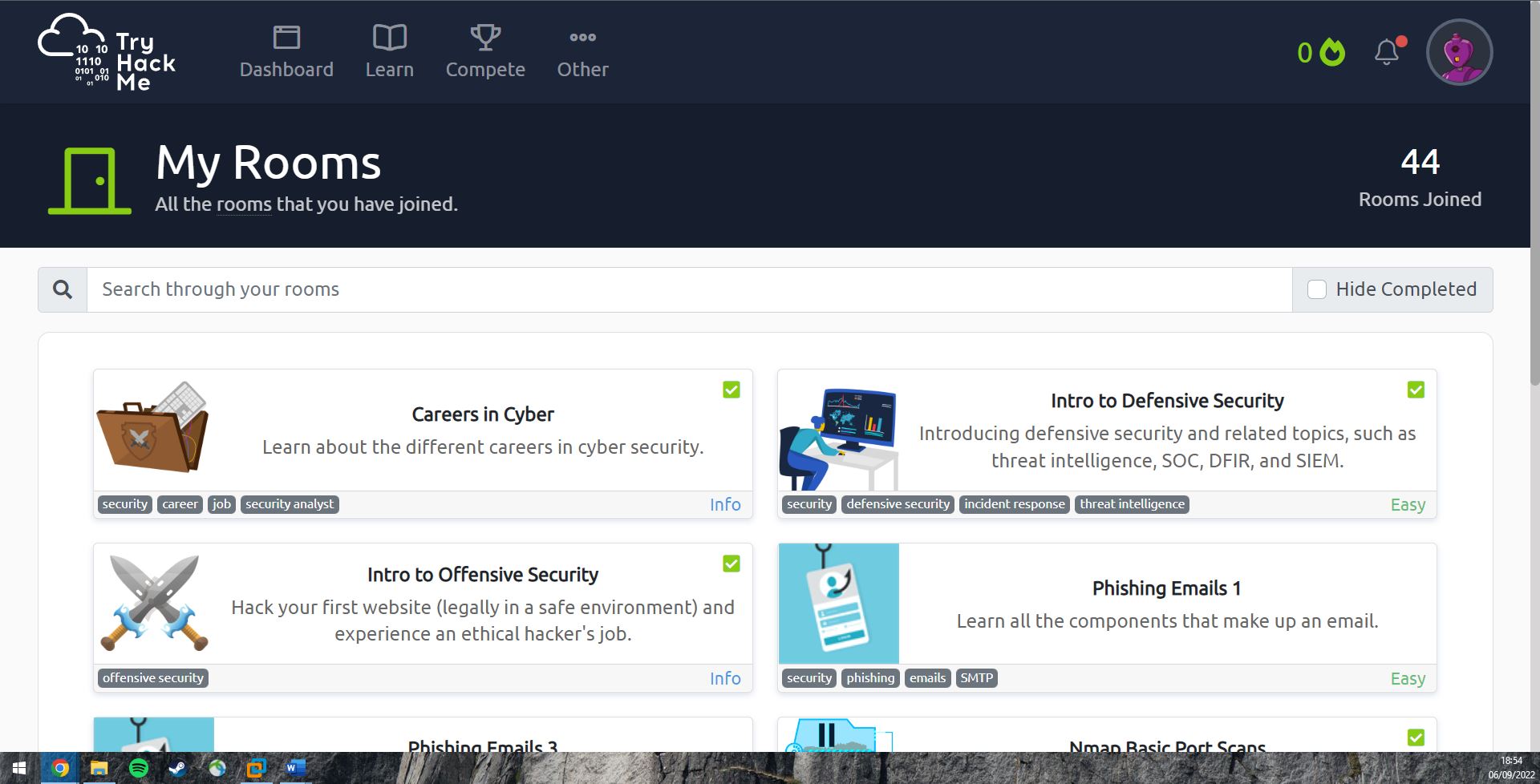
# INTRODUCTION

This report will take you through my learning process and semester progress for the Fontys, ICT, Cyber Security (S4) course.

My name is Mohammed Alshukaili. Although I come from media design background, my passion is to study cyber security. I consider myself an absolute beginner in this field. I will do whatever it takes to become a good student throughout the whole semester.

# IT Background

I only studied infrastructure engineering for 10 weeks in my first semester at Fontys. I felt like this was not going to be enough to become at a high-level student. Therefore, I decided to invest all my summer breaks studying new things that are related to infrastructure.

For instance, I subscribed to [www.tryhackme.com](http://www.tryhackme.com), a website that provides a huge amount of IT courses. I finished all the fundamentals for Linux, windows, and online databases. 

This picture shows some of the rooms that I enrolled for. I finished most of them with so much information gained.

I also like to learn by watching sometimes. I am subscribed to so many IT YouTube channels that provide basics for the IT industry. Some of my favorite YouTube channels are

* <https://www.youtube.com/c/PowerCertAnimatedVideos>
* <https://www.youtube.com/c/H>

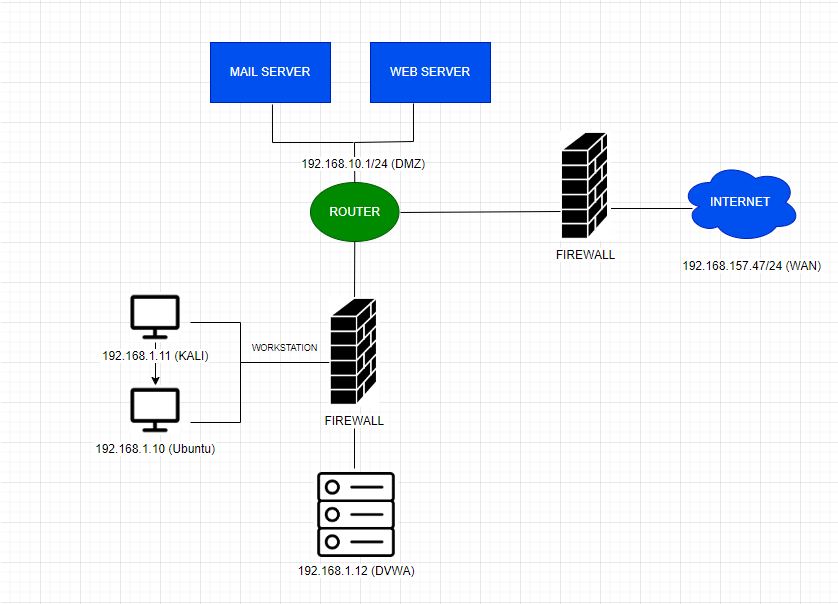
What I realized after all the tutorials that I have been to is that I prefer reading rather than watching tutorials. As the visual tutorials usually do not give me the chance to do things myself. On the other side, services like TryHackMe, Fontys are more practical that I found myself enjoy it more.

# (BOK) Web Application Security (week 1-4)

(picoCTF)

The second day, I attended the networking refresh. We learned about vSphere and how to set up a local virtual network. I thought that we would need to borrow some routers and devices to set up local network. However, the VLAN concept really changed the way I see internet. I managed to design a local network that includes 2 firewalls, workstation, DMZ, and the WAN to access the internet.

Here is a picture of my simple design:



I had to learn pfSense to set up this network and some of Docker basics.

Last class of the week, we had a theory about security and the different types of attacks.

## Path Traversal

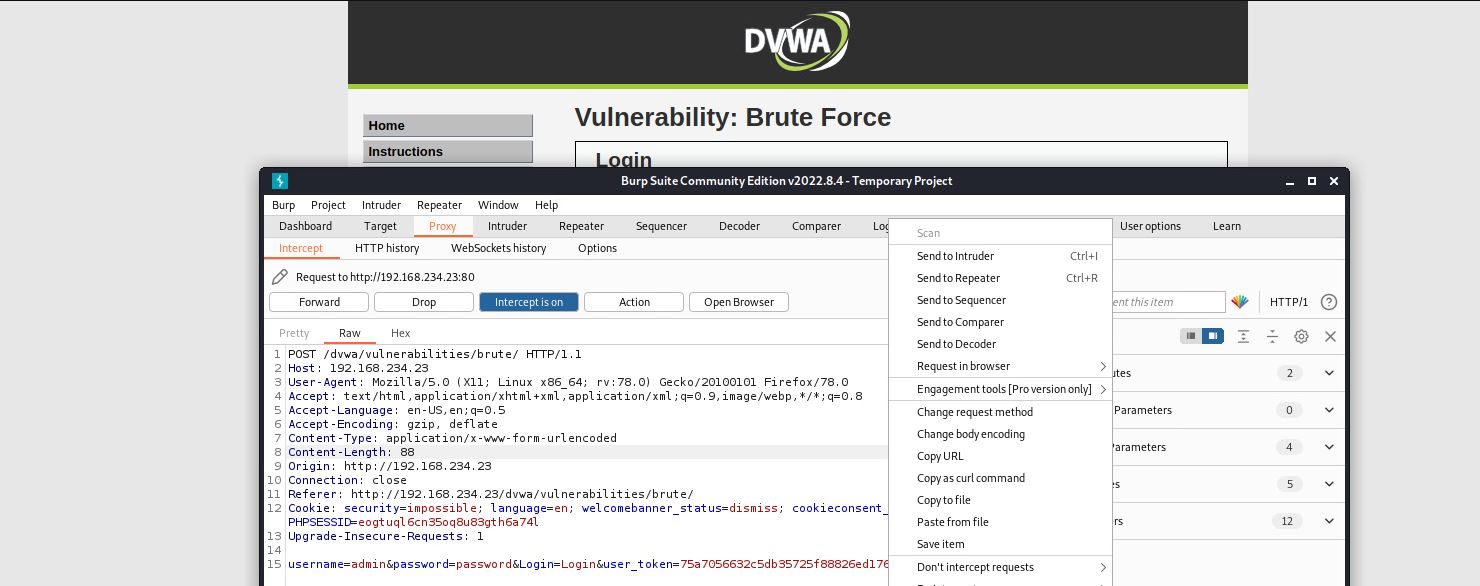
Moving to the real pen testing, we started the second week with a class for path traversal. It is when you study what is a specific system and then be able to know the file names and where sensitive data might be stored, you can navigate to the system files until you reach some private data.

Also, you could get to trick the terminal that you are someone else and execute commands on behalf of them.

## Brute Force

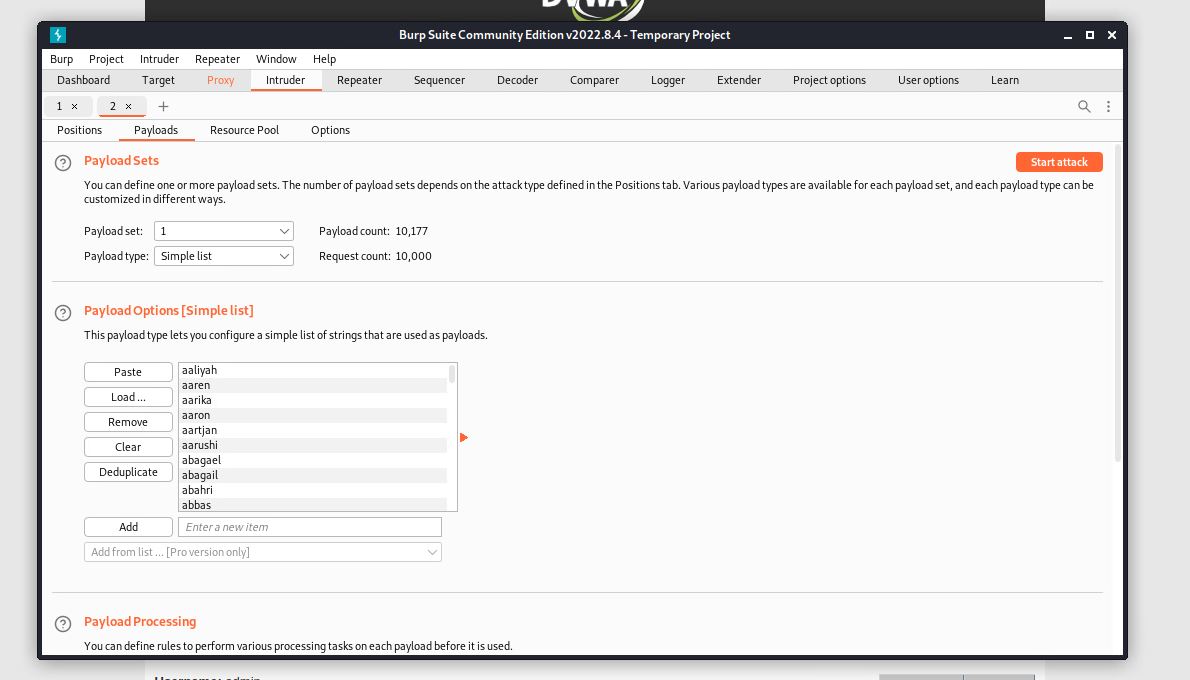
Brute force is when the attacker guesses the username/password to log into places they do not have access for. The way I would do this is by sending the POST request to Burp Suite to adjust the inputs as I desire. Burp Suite has many methods that have several purposes. For instance, Intruder is to specify a payload that you want to try so many inputs in that payload, and when burp suite returns a 200 status for the input (what the attacker guessed), means that the input was successful. Other way to know that the input is successful is by looking at the length as the correct input differs from the others.

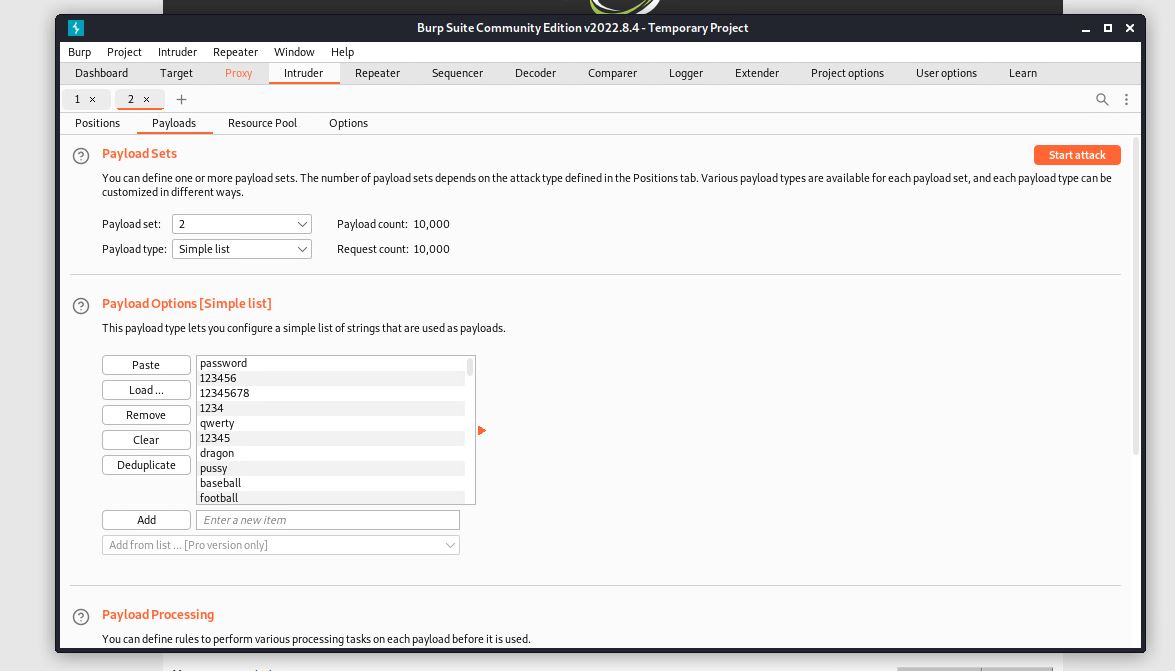
There are hundreds of username/password files that are available to make it easier for the pen tester to use instead of having to type the common username/passwords, you can just use these wordlists(/usr/share/secLists/), or you can find many other files on the internet.

Let’s get back to brute forcing, when capturing the request on Burp Suite, we should right-click and “Send to Intruder”. 

Then navigate to Intruder tab to specify the payloads that you want to test. For example, $username$(payload 1) and $password$(payload 2).

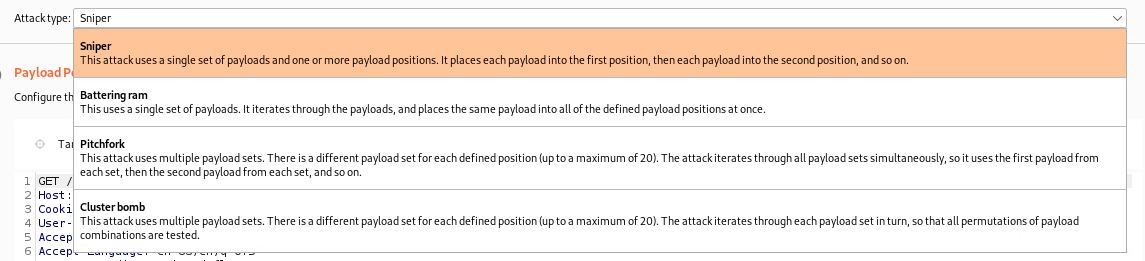
Start the attack and when there is a correct payload you would know by comparing to other inputs.





There are many Intruder methods. Such as, Sniper, Battering ram, Pitchfork, Cluster bomb.

You can choose one of them considering you situation.



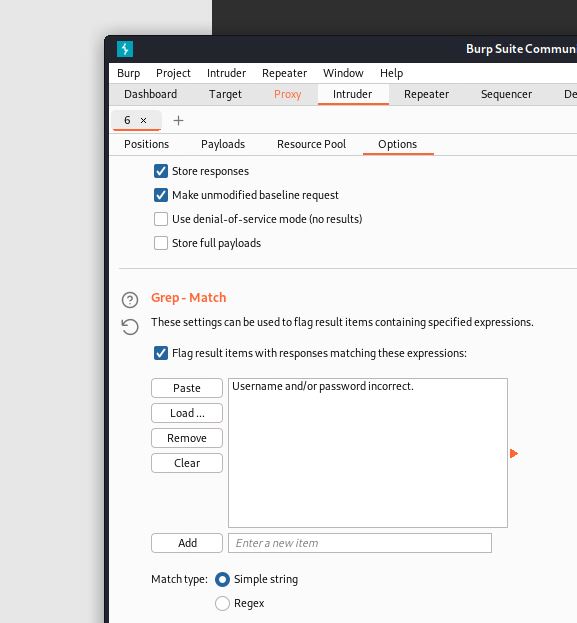
DVWA Exercise

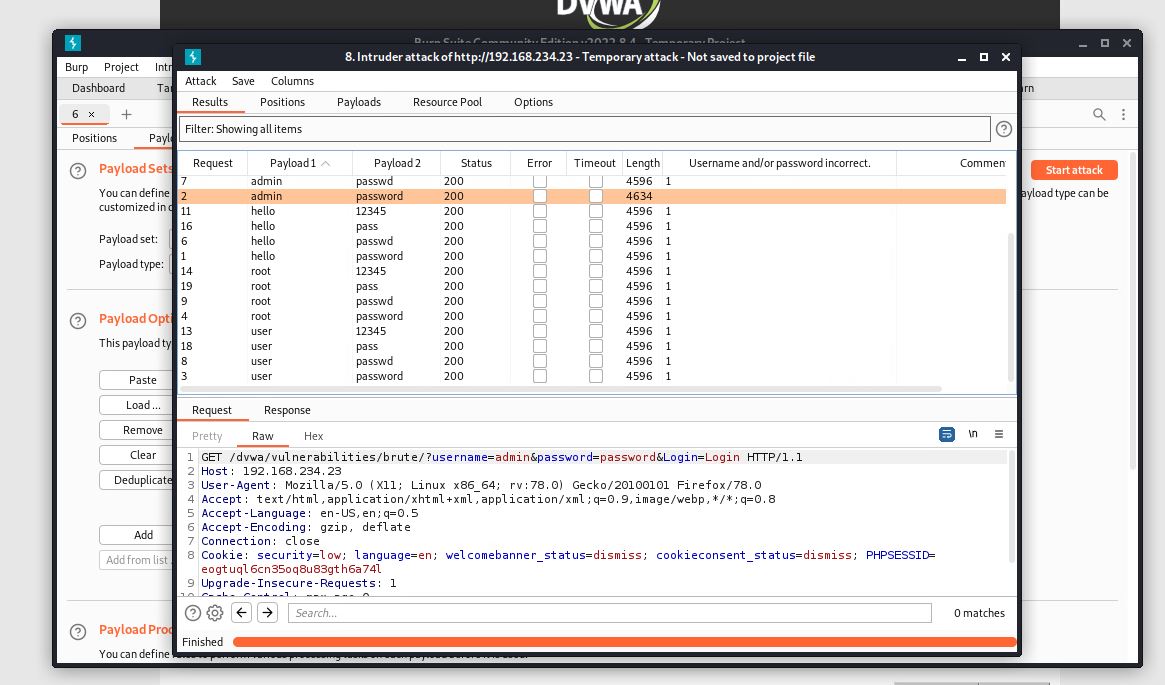
### Brute Forcce – Low level

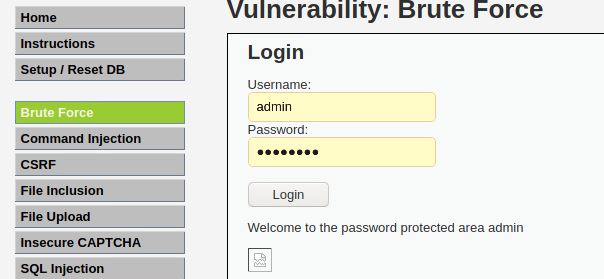
The correct answer is:

Username: admin

Password: password

You can notice it is the one that differs from others. I decided to filter the outputs as if they cause error and the correct answer is the only option that does not. 



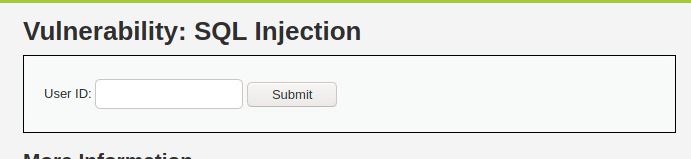


## SQL injection

This method attacks the database by tricking them to returning sensitive data. Websites have become more interactive than before. Therefore, the user is expected to manipulate the programming language code. For example, when the user is typing something in a web input, the website is ready to take that value to execute it to the website. This is a chance for the attacker to input malicious code that might affect the website in a bad way. One of the SQLi tricks is to force the database to return all the information by putting “1=1” in the sql query. 1=1 means that the query is always true, so it returns all the data.

DVWA Exercise:

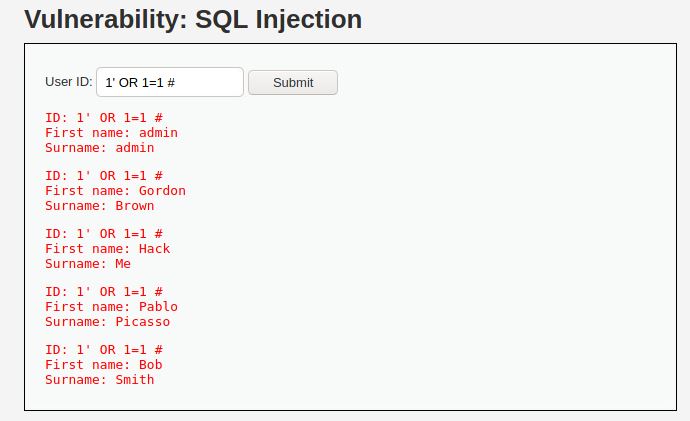
### SQLi – Low level



The website here expects a number as input to return data that are true to that specific number. However, we can manipulate the query but first we need to understand it.

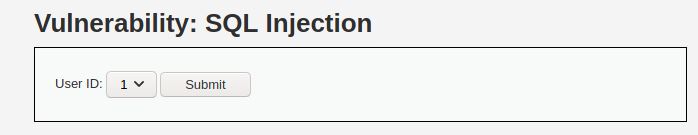


Simple query that we can easily adjust by entering a number as it asks, then close the quotation mark then type “OR 1=1” as it would return everything as true. Then finish the query with # to comment anything comes after.



Indeed the database returned all the data because of the 1=1 logic.

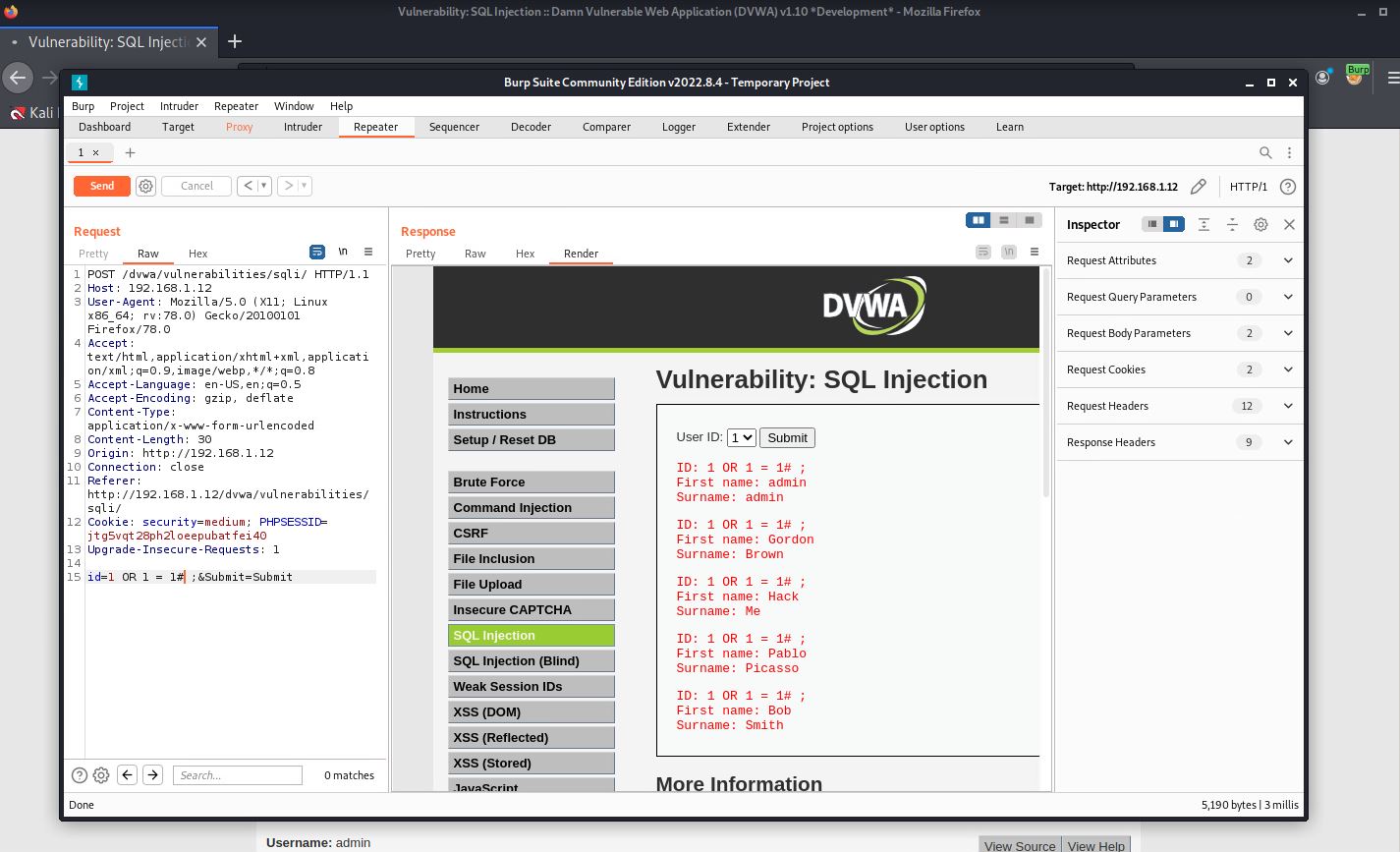
### SQLi – Medium level



The difference here is that we are not given input to enter true logic with the input. We are just asked to select the number from a drop down. The idea for how to do this is still exactly like the old one. However, we just need to find a different way to put the logic onto the database.

For this I would choose Burp Suite to help me manipulate the POST request.

I just need to send the POST request to Burp Suite then send it to repeater to execute my logic there.

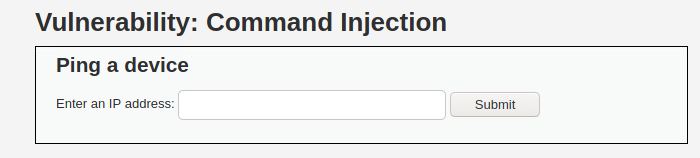


As you can see it is similar to the previous one, we only had to find a way to manipulate the POST request.

## Command Injection

### Command Injection – Low level

We are presented with an input to enter any ip address to PING.

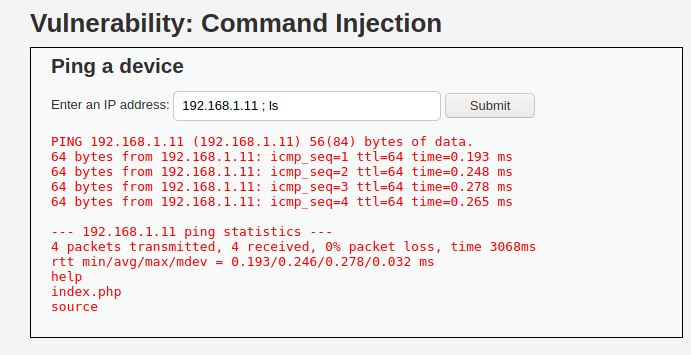


Let us see the code behind this function:



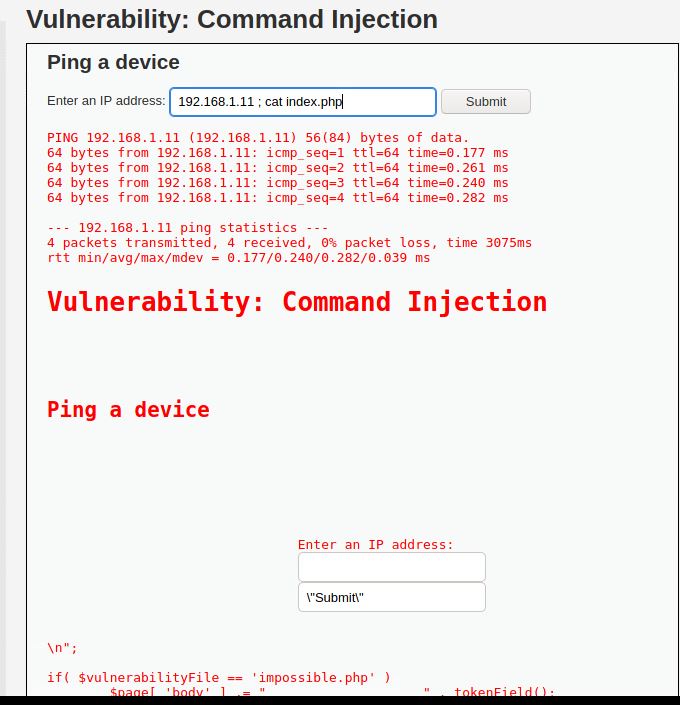
This function basically tells the program to go to the cmd and ping the target (entered ip).

It should be too easy to tell the input to do more commands. For example, we can enter any ip normally then we can finish the command by putting ; in the end. After that, we can enter more commands to get more details like (ls, pwd, etc).



As we ping the ip, we also managed to execute ls command which outputs files and folders in the system (help, index.php, source)

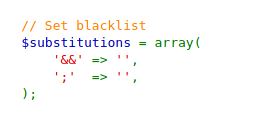
Let us see the content of index.php for example:



### Command Injection – Medium level

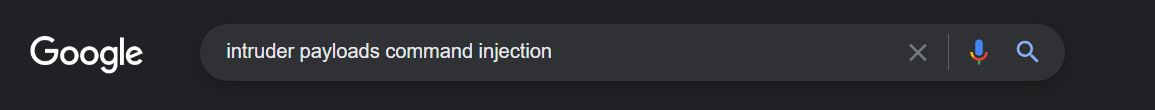
When trying to do the previous method here you would not get any response.

Why? Looking at the code:



We see that some symbols are not allowed anymore. Good but not good enough as the attacker can easily to adjust the POST request in Burp Suite to look for the allowed symbols.

First I went to google.com and looked up:

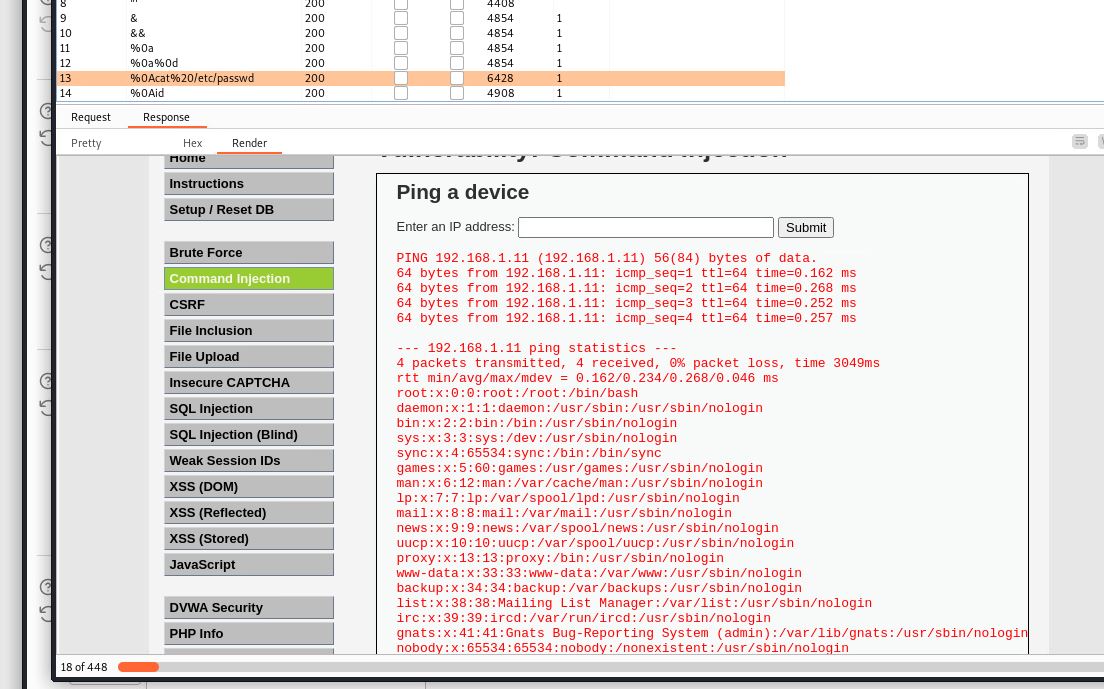


I found this page

<https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Command%20Injection/Intruder/command_exec.txt>

This github page has so many command injection payloads that I can try for the exercise.

I sent the request to Intruder. Sniper method helped me to load the words list and I filtered the output to packet loss as I would know the right answer If the packet loss check box was checked.



Eventually, I got to output the passwords file.

# (BOK) Network Security (week 5-8)

# (BOK) Security Concepts (week 9-10)