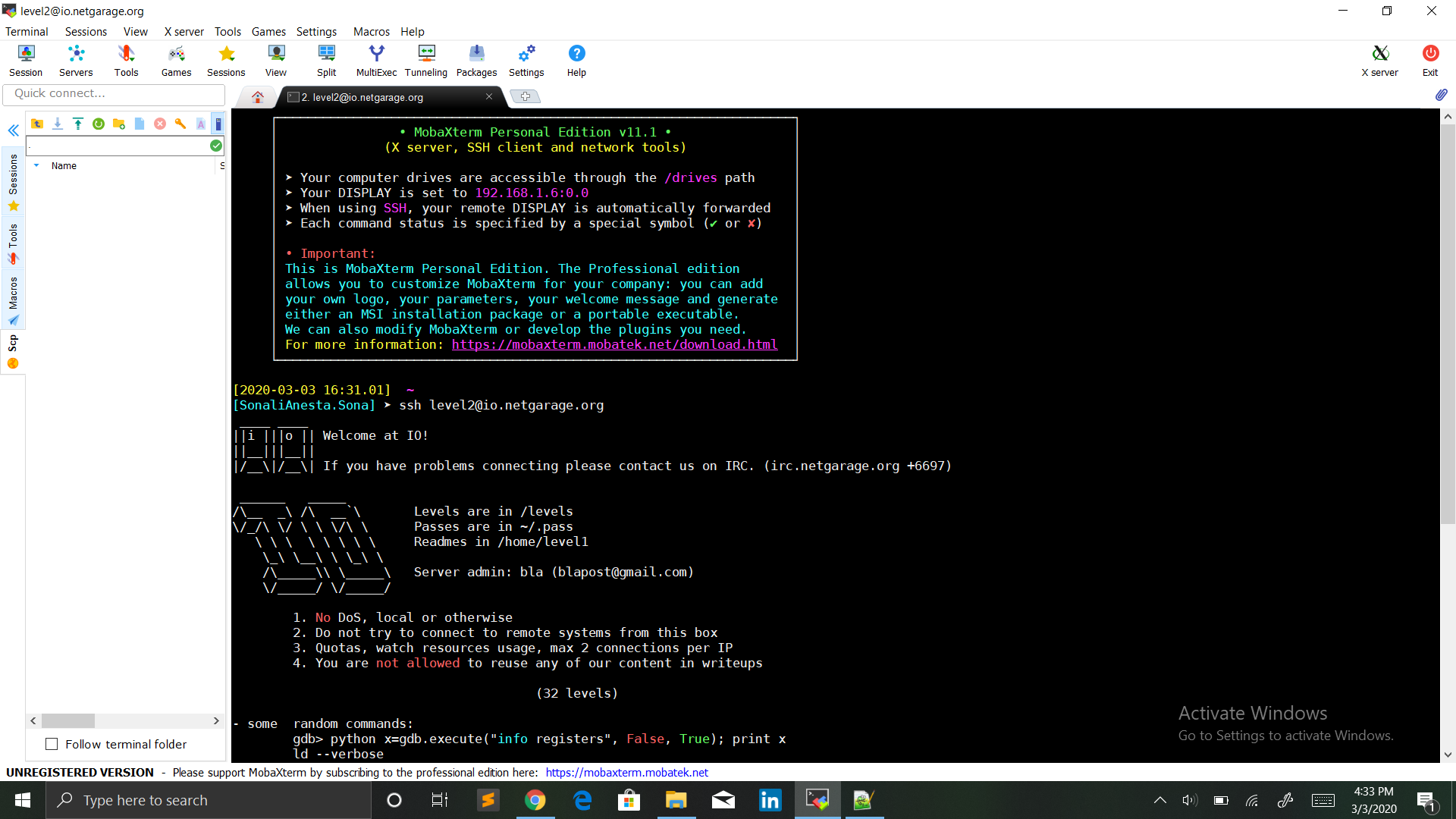
IO.NETGARAGE.ORG

LEVEL 02

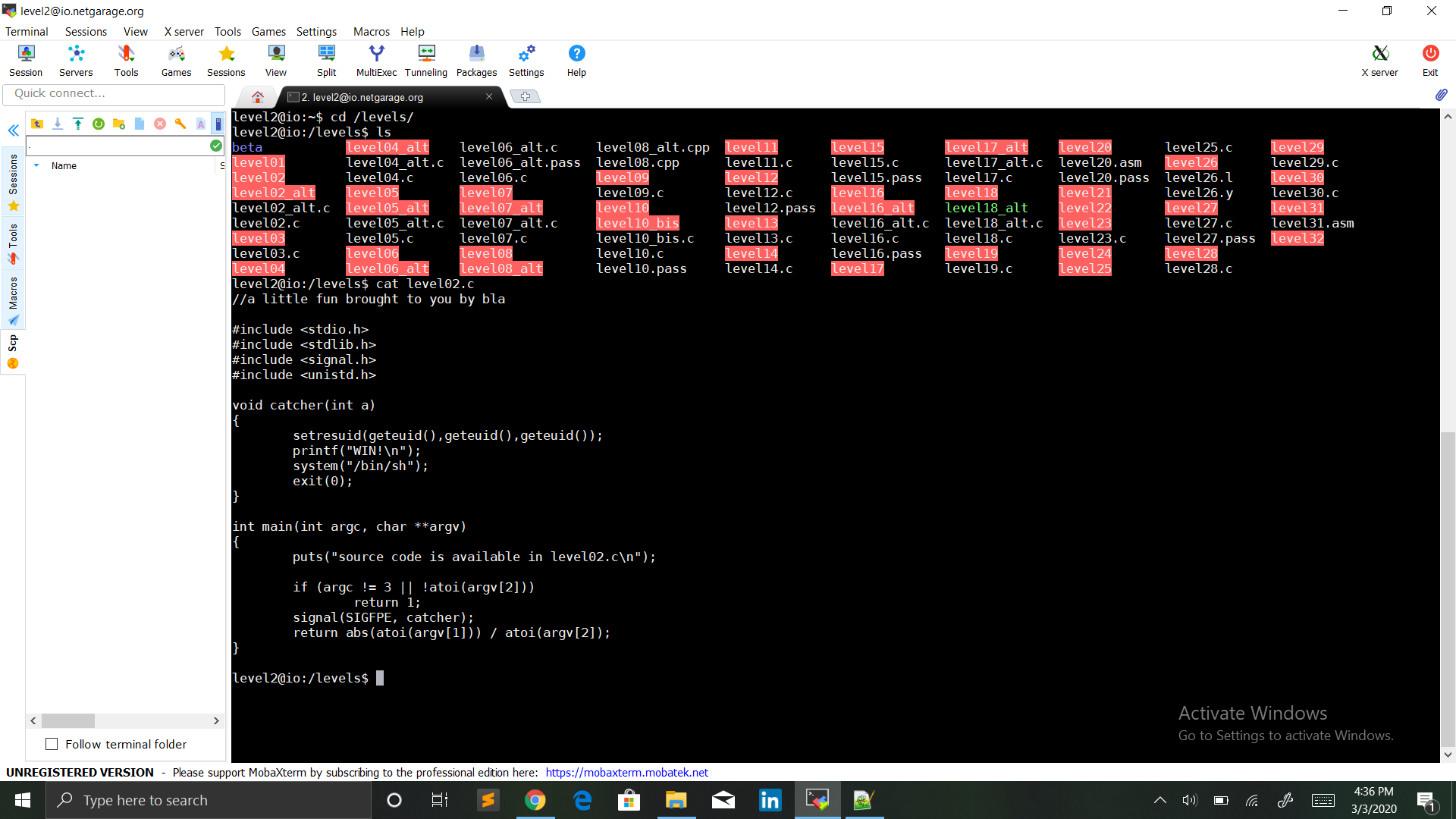
IT17098724

Anesta W.D.S

* Using link and password log in to the wargame level 02 as shown below.



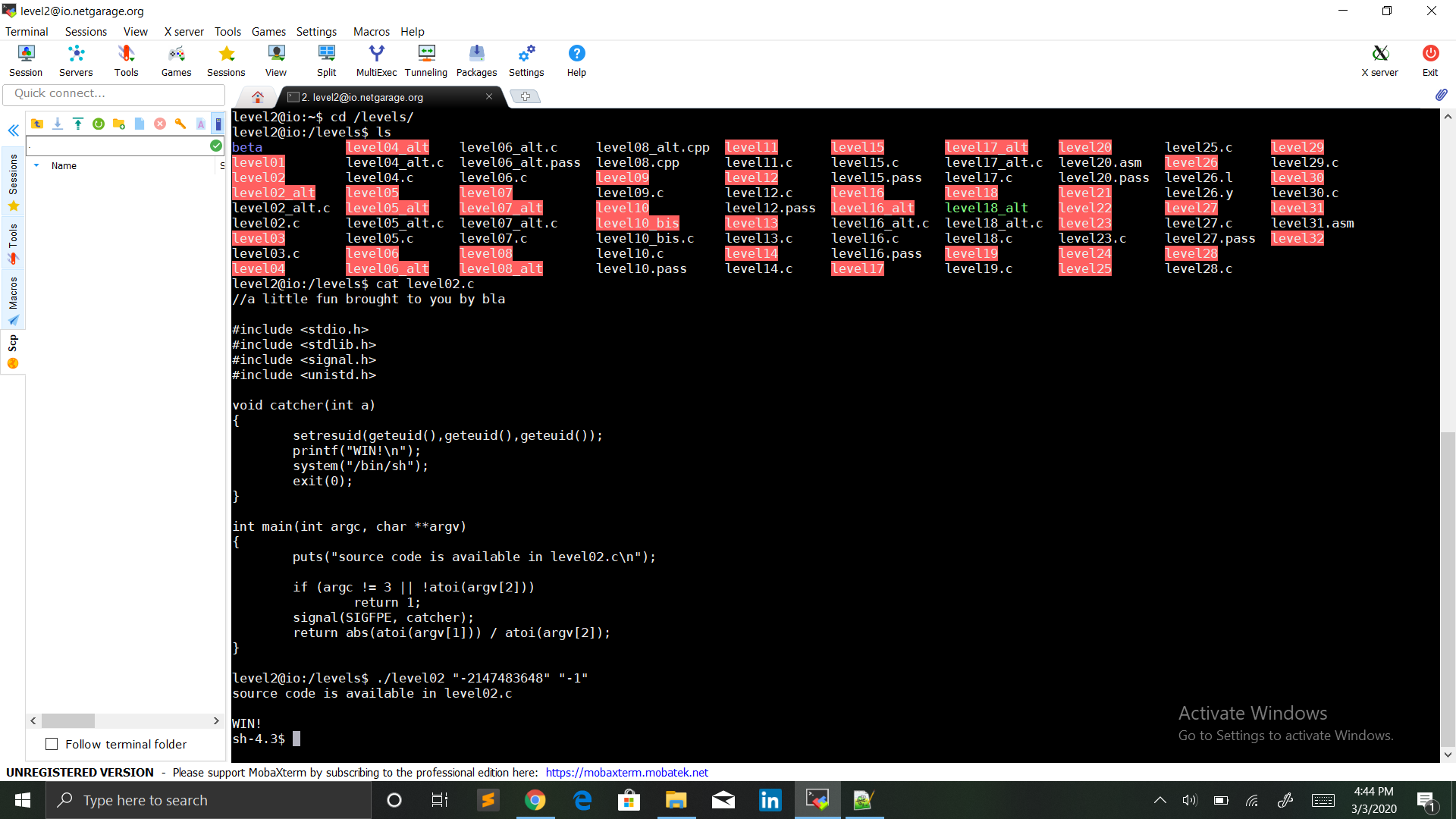
* Open the level02.c file



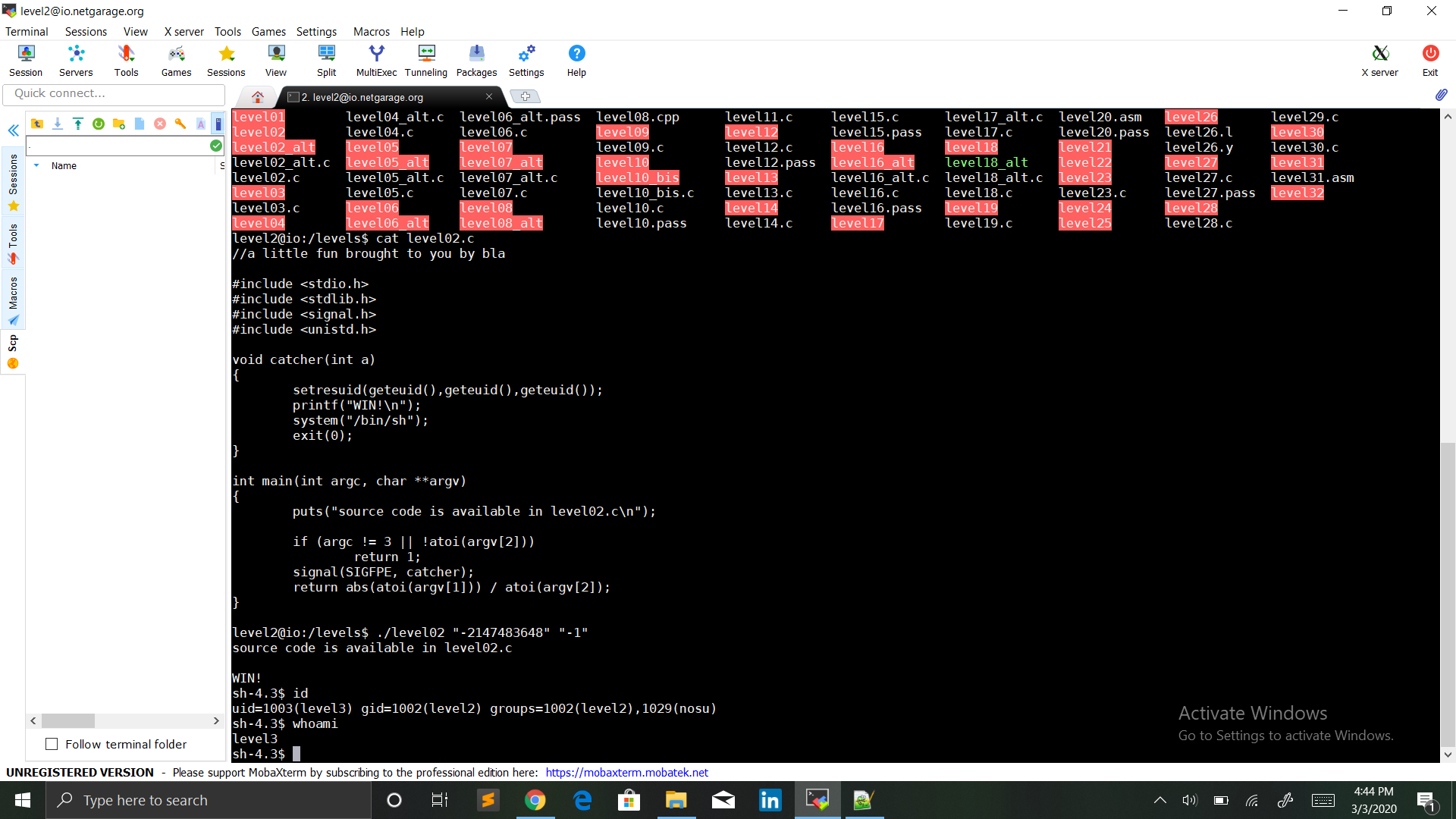
* Opening the level02.c file we can see:

1. The number of **args** must be 2, (**argv[0]** being the caller's name).
2. The two arguments should be numbers
3. The catcher function will be called on the event SIGFPE (launched for example for a division by 0)
4. The return value of the function is **argv[1]**/**argv[2]**

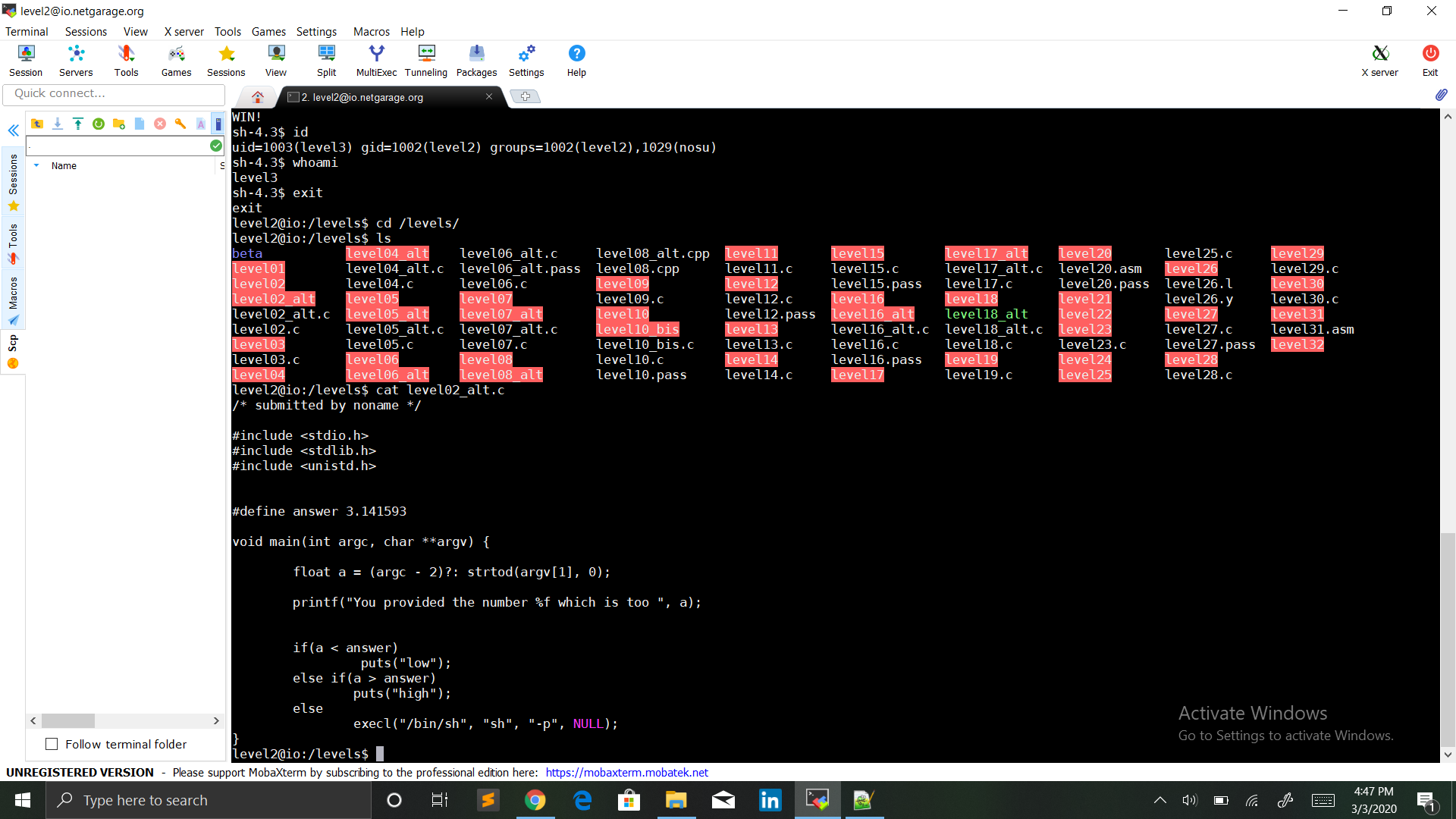
* If the catcher function is called, it will set the current user identity, and print a winning message, before spawning a new shell. This is clearly the indication that we need to raise a **SIGFPE** exception.
* The **SIGFPE** can be triggered with a 1/0 or an sqrt(-1) for example, in our case, neither can be used.
* What we can do instead is try to use an integer value outside of the bound of the integer definition. We can see on the **abs** that the most-negative value to be out of range is -2147483648 because this will convert to 2147483648, above the **MAX\_INT** value. So, if we send to abs the value - 2147483648, the result will be also -2147483648 (because of binary max bound and negative values).
* But, if we send the value 2147483648 to **abs**, which is an incorrect value, it should raise an error:



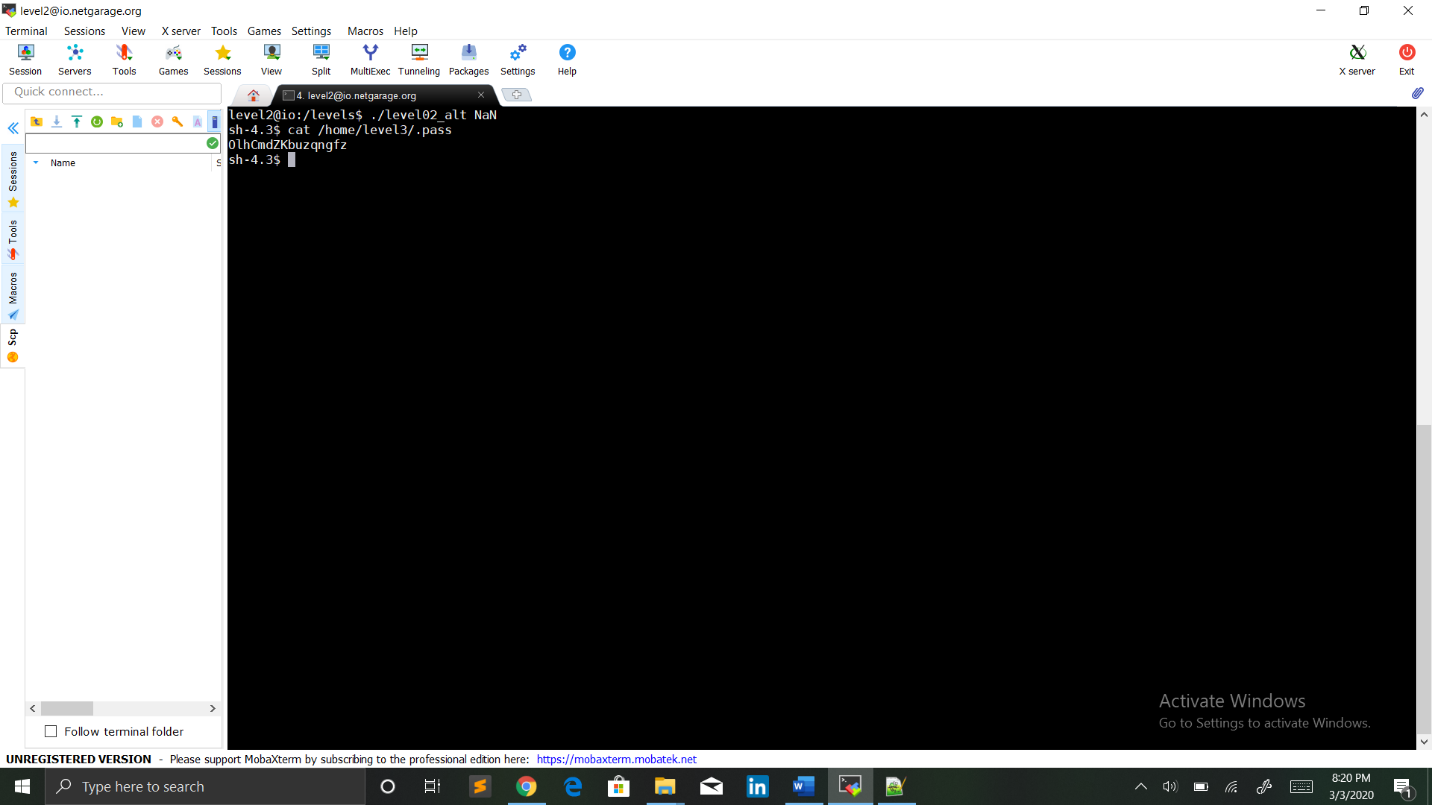
* We successfully obtained a shell. But we can see that effective user-id not there.



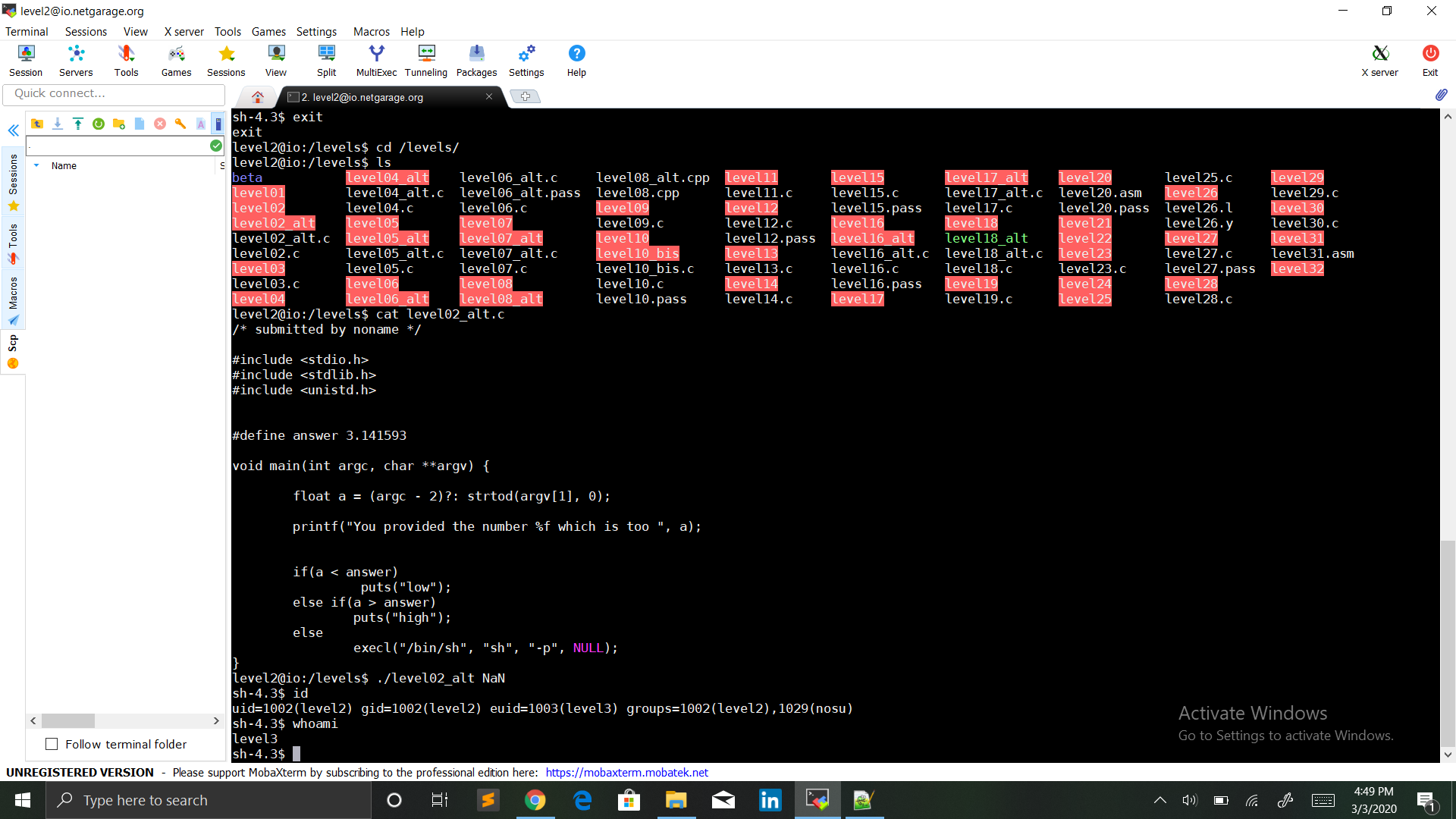
Then go the level02\_alt.c file

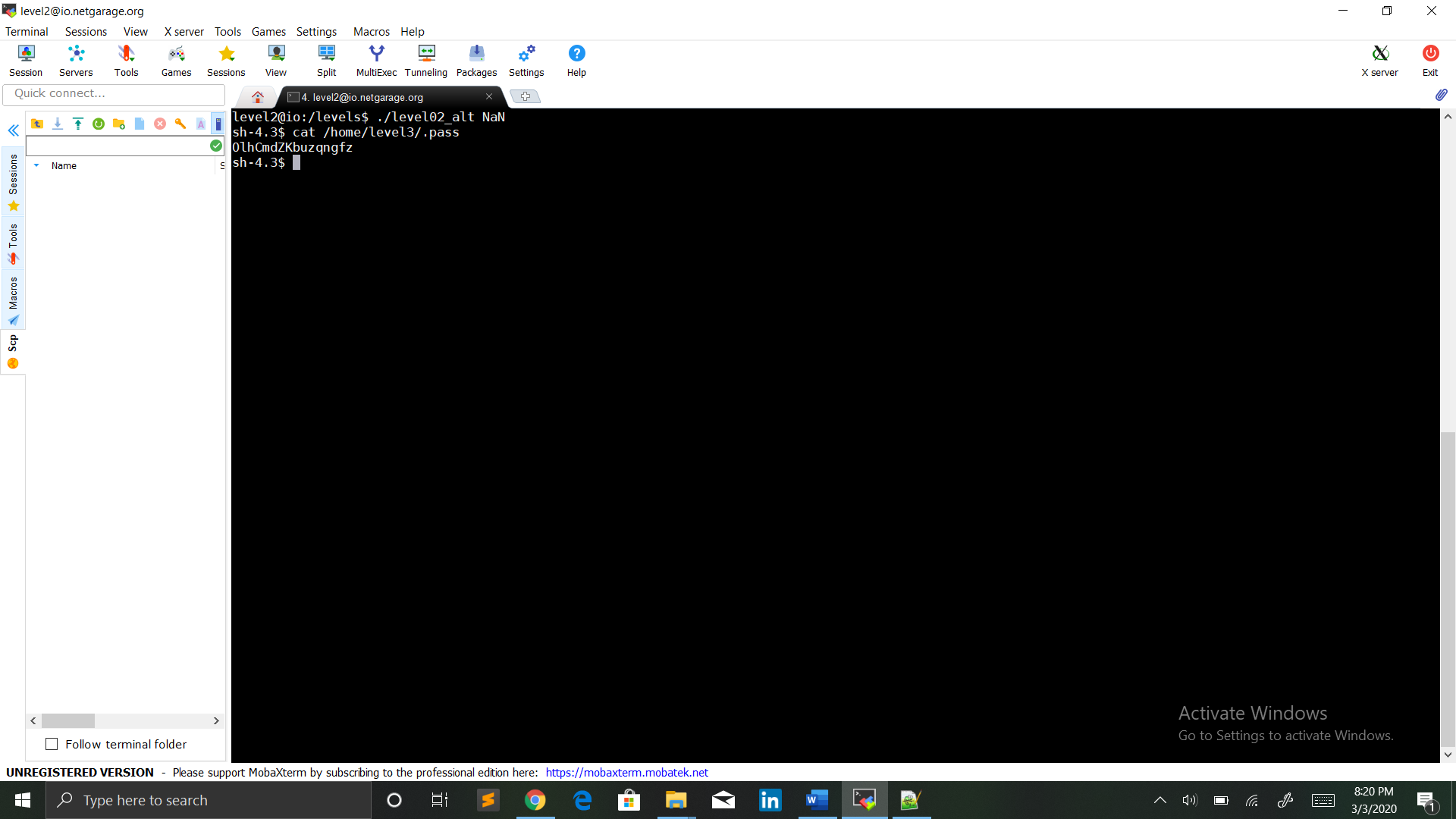


* Expect keying in 3.141593 to give you an easy shell. However, the trap is that this is a floating-point number. In computer science, comparison with a floating-point such as == can be very dangerous because it can never be true.
* To solve this challenge, you need to recognize that you have to fail the first 2 conditions ( a < answer ) and ( a > answer ).
* Turns out that **strtod** allows one to specific special tokens like INF for infinity and NAN for not a number. By specifying NAN, both conditions will fail regardless hence allowing us to go into the else function.



* We successfully obtained a shell. We can see that our effective user-id became level 3. Thus, we have access to the password for the next level.





Level 3 password: **OlhCmdZKbuzqngfz**