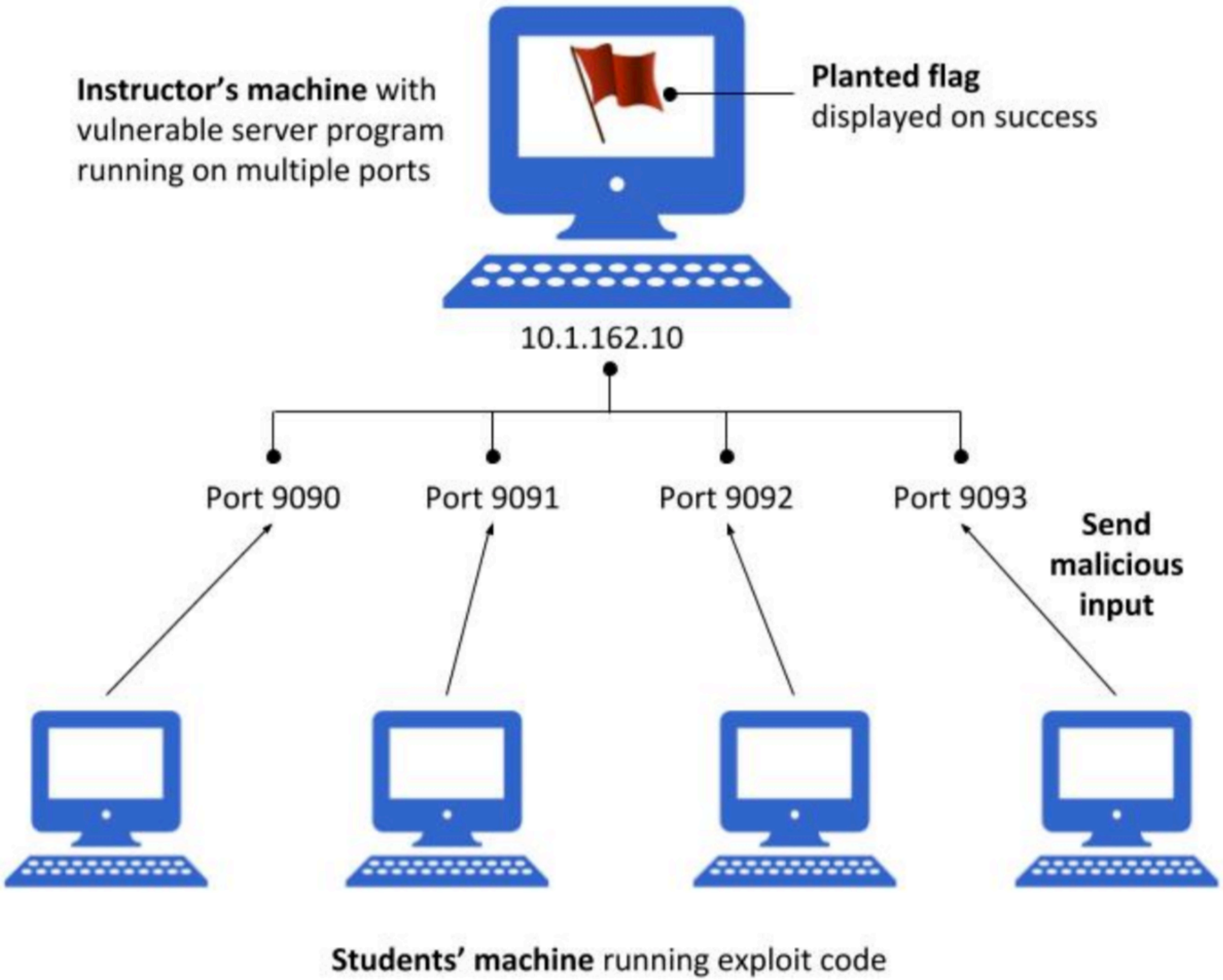
Introducing the Buffer Overflow Attack CTF!

All of the resources you will need are located on GitHub:
https://github.com/traviswpeters/csci476-code/tree/master/CTF_buffer_overflow

Goals

- Applying what you've learned to a slightly new setting
- Navigating a more complex environment to carry out a more realistic attack
- Working as a team to problem-solve
- Experience a CTF-style competition

CTF Competition Setup *(example)*



Instructor's machine with vulnerable server program running on multiple ports

Planted flag displayed on success

10.1.162.10

Port 9090

Port 9091

Port 9092

Port 9093

Send malicious input

Students' machine running exploit code

*YOU MUST BE ON **MSU-SECURE***

Levels

	Buffer Address	Buffer Size
Level 1	exact value	exact value
Level 2	exact value	range
Level 3	range	range
Level 4	none	none

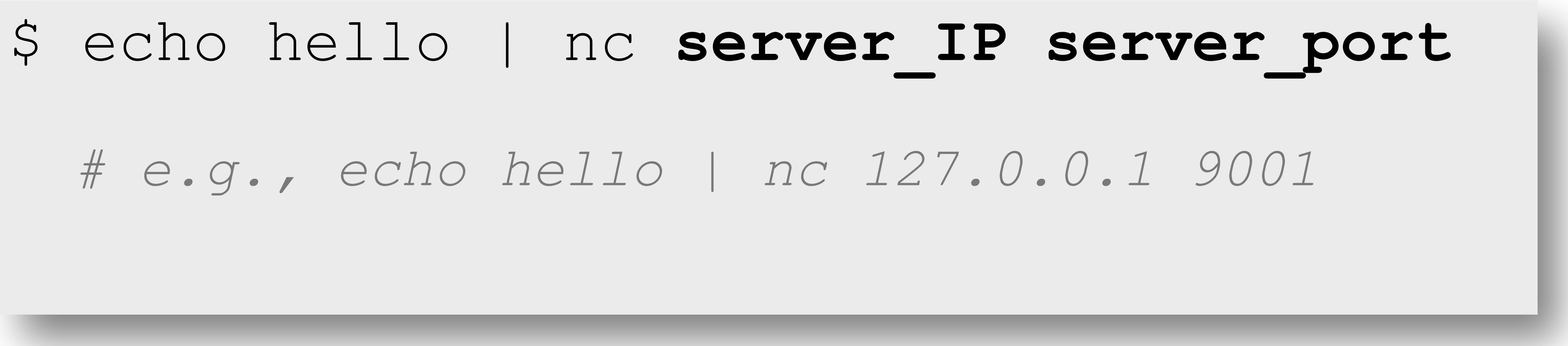
IP ADDRESS = **10.152.183.104**

PORT = **90[##]**

e.g., 9001, 9009, 9015, 9022, …

Sanity Test…

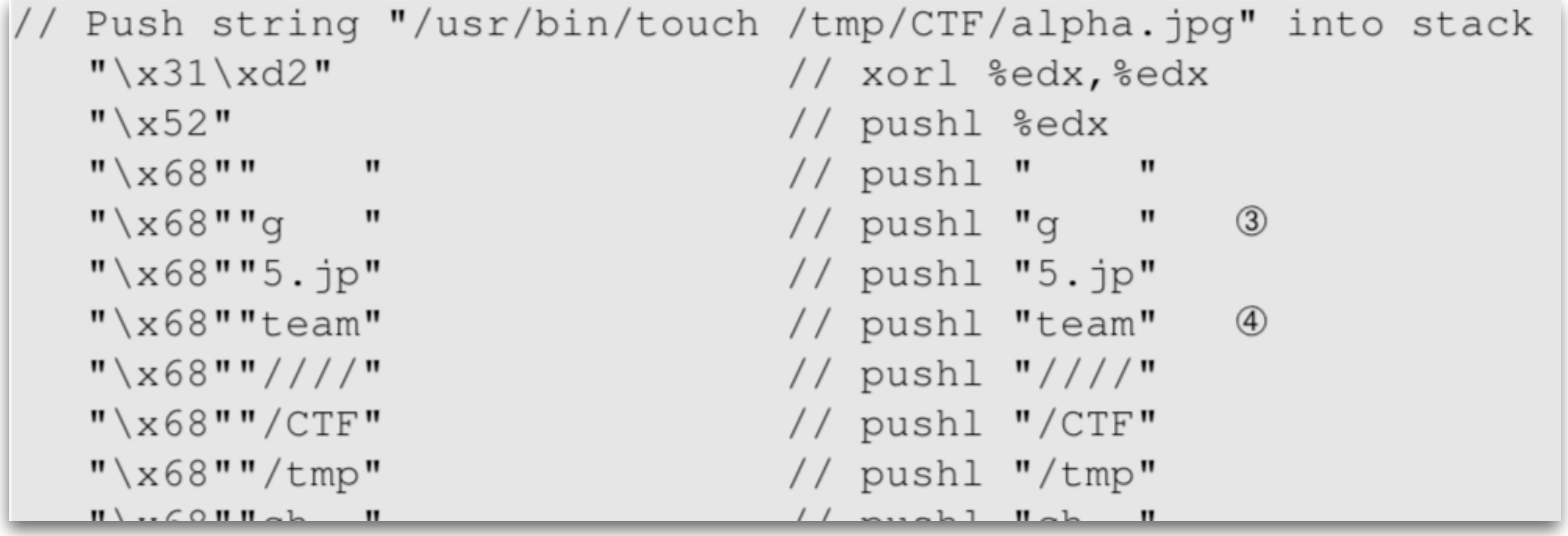
(before starting each level…)



```
$ echo hello | nc server_IP server_port

  # e.g., echo hello | nc 127.0.0.1 9001
```

ALSO, NEED TO UPDATE FLAG NUMBER IN **EXPLOIT.PY** BASED ON YOUR TEAM NUMBER



```
// Push string "/usr/bin/touch /tmp/CTF/alpha.jpg" into stack
   "\x31\xd2"                       // xorl %edx,%edx
   "\x52"                           // pushl %edx
   "\x68""    "                     // pushl "    "
   "\x68""g   "                     // pushl "g   "   ③
   "\x68""5.jp"                     // pushl "5.jp"
   "\x68""team"                     // pushl "team"   ④
   "\x68""////"                     // pushl "////"
   "\x68""/CTF"                     // pushl "/CTF"
   "\x68""/tmp"                     // pushl "/tmp"
   "\x68""ch  "                     // pushl "ch  "
```