CSE 351: Week 6

Tom Bergan, TA

Today

- Questions on the midterm?
- Lab 3

Lab 3: Buffer Overflow

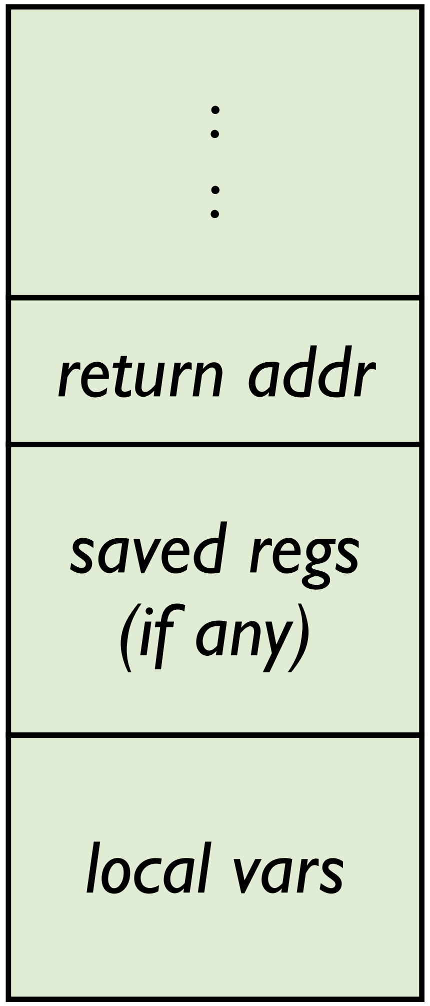
This has a buffer overflow

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

Why?

- `Gets()` doesn't check the length of the buffer

The Stack in getbuf()

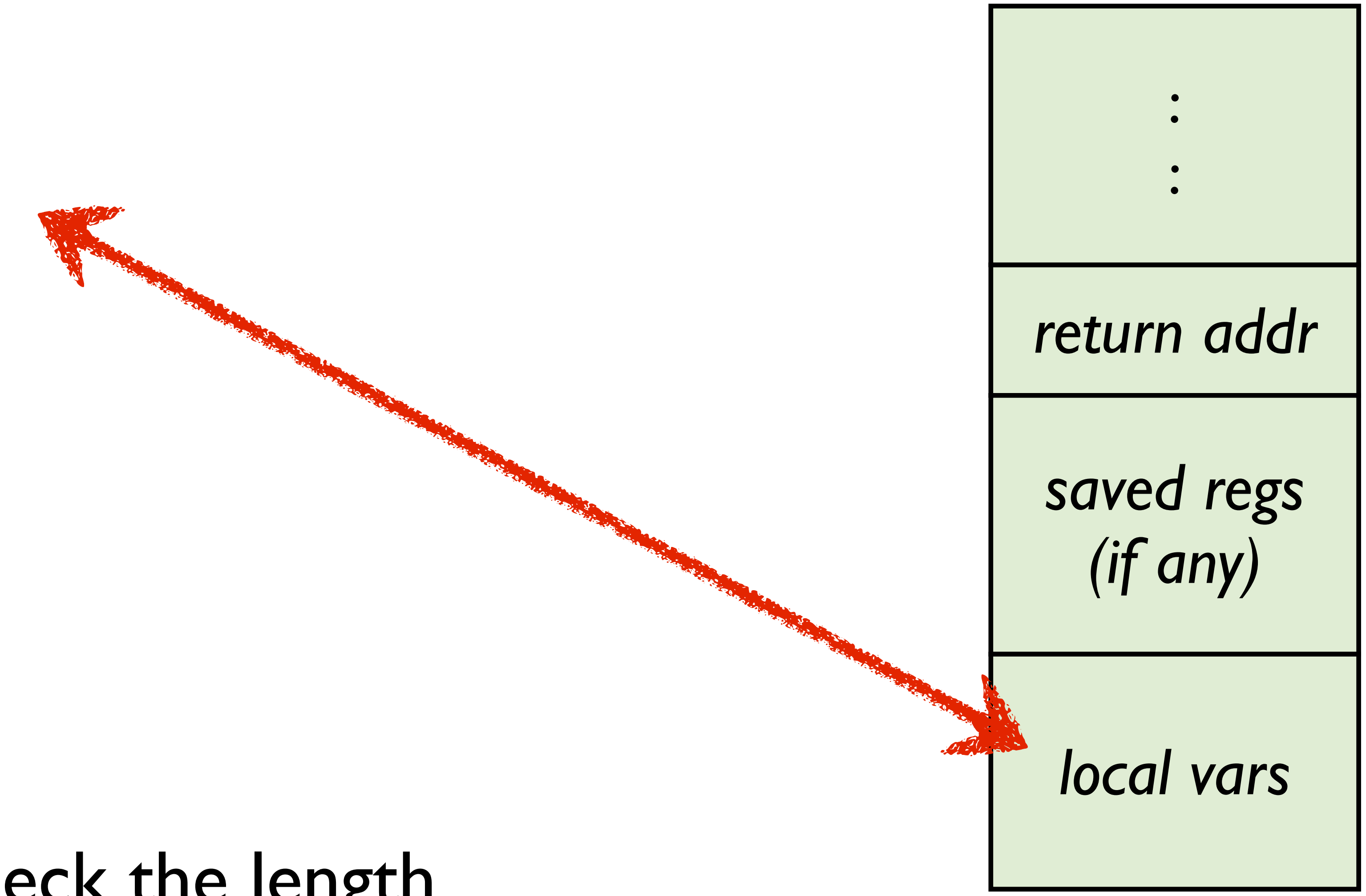


Lab 3: Buffer Overflow

This has a buffer overflow

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

The Stack in getbuf()



Why?

- `Gets()` doesn't check the length of the buffer

Lab 3: Buffer Overflow

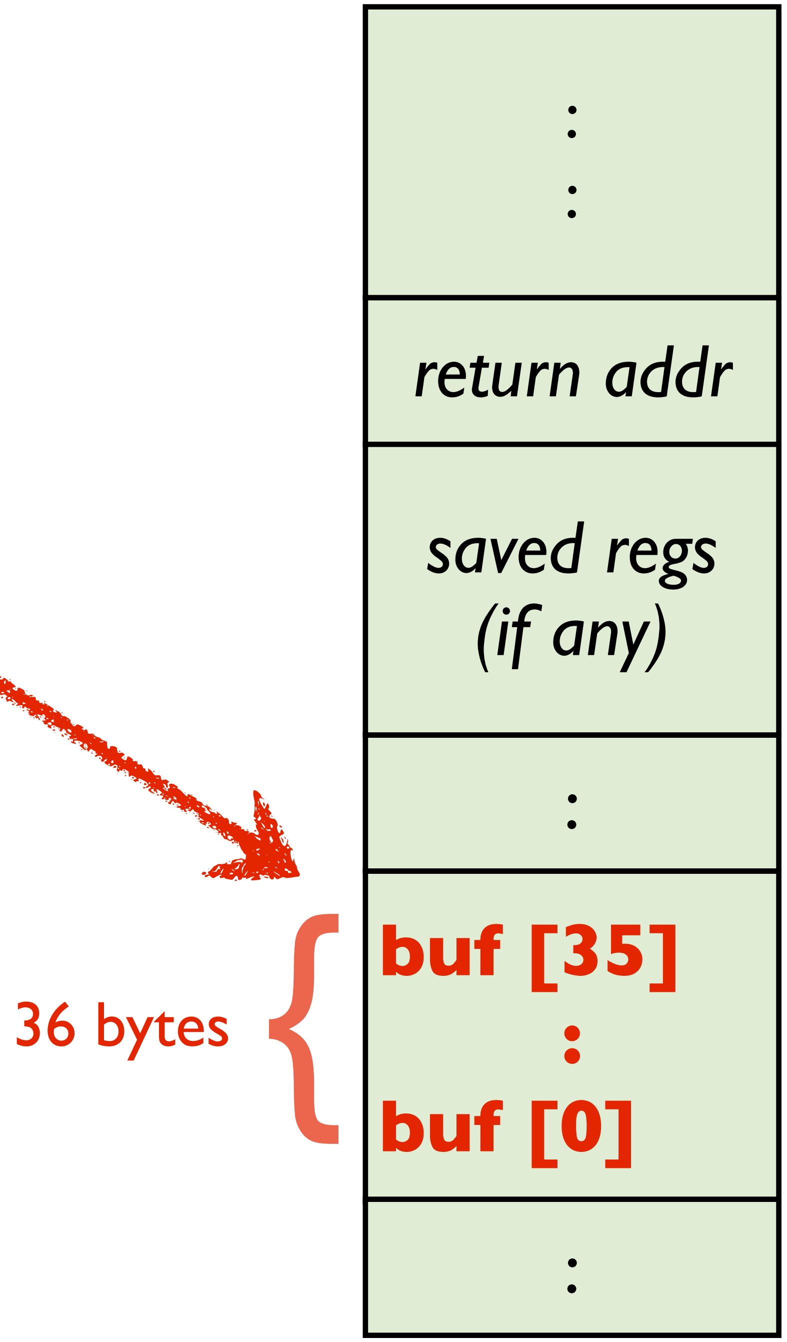
This has a buffer overflow

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

Why?

- `Gets()` doesn’t check the length of the buffer

The Stack in getbuf()



Level 0: Call `smoke()`

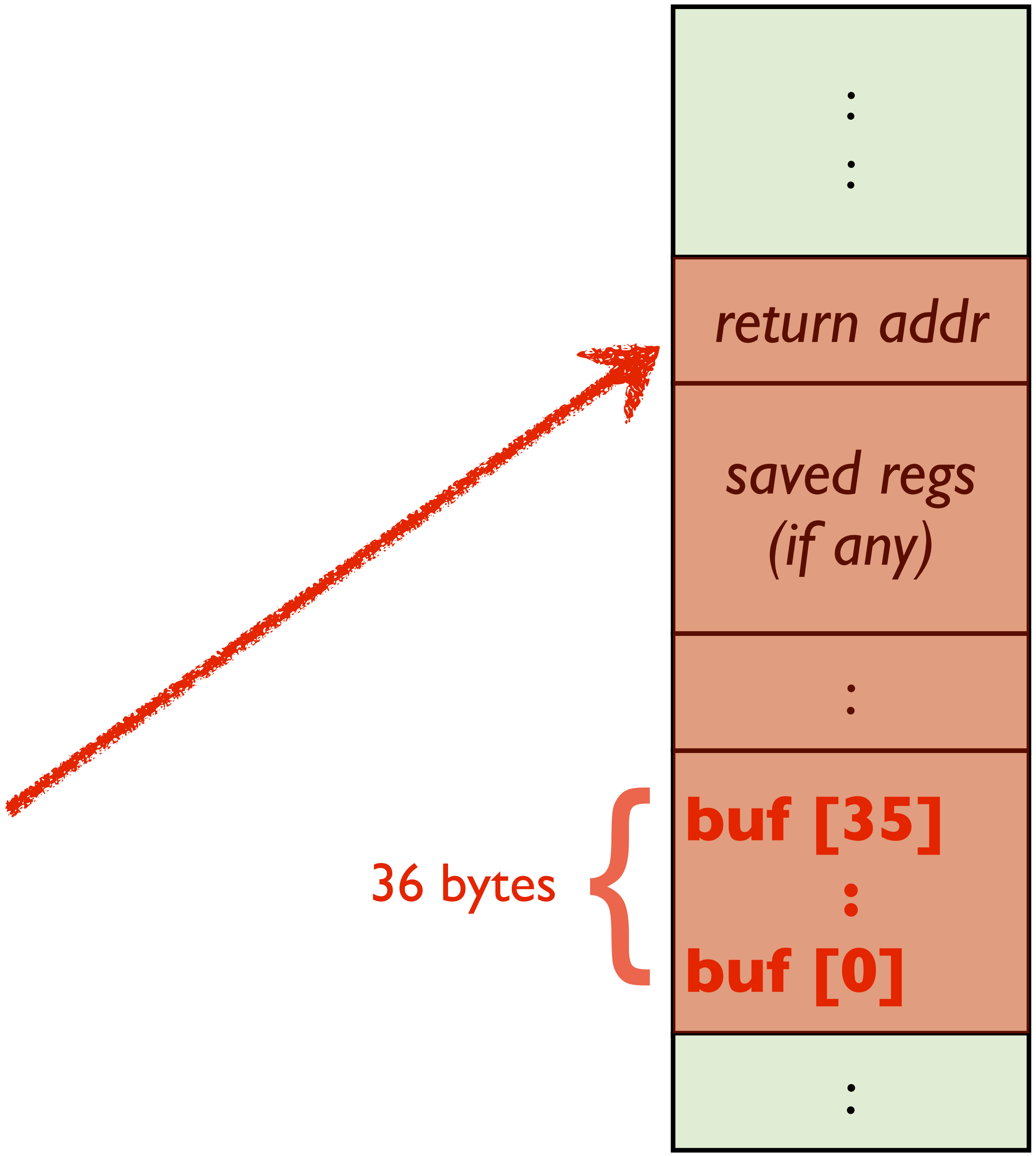
Goal: call the smoke() function from getbuf()

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

How?

- overwrite the return address so we "return" to smoke()

The Stack in getbuf()



:

:

return addr

saved regs (if any)

:

buf [35]

:

buf [0]

36 bytes

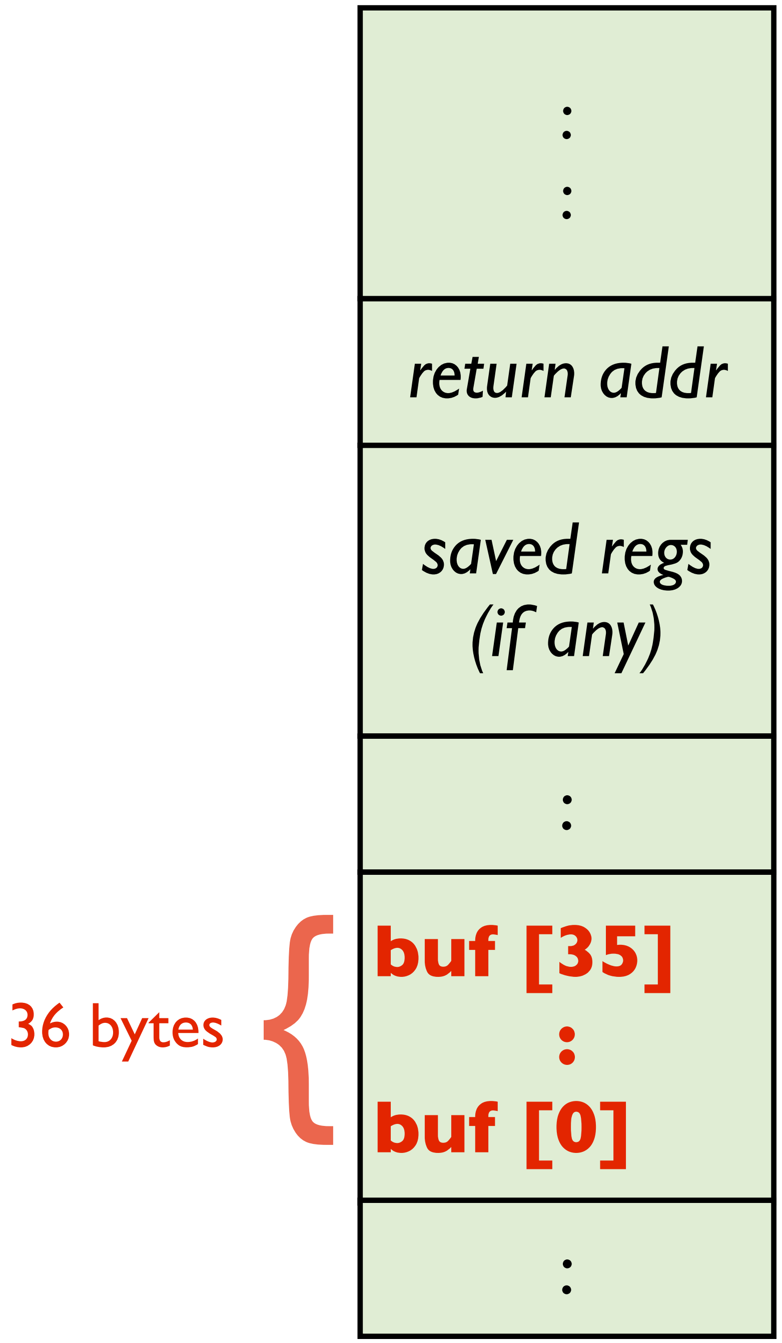
:

Level 1: Call `fizz()`

Goal: call fizz() with a special parameter (your "cookie")

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

The Stack in getbuf()



:
:

return addr

saved regs (if any)

:

36 bytes

buf [35]

:

buf [0]

:

Level 1: Call `fizz()`

Goal: call fizz() with a special parameter (your "cookie")

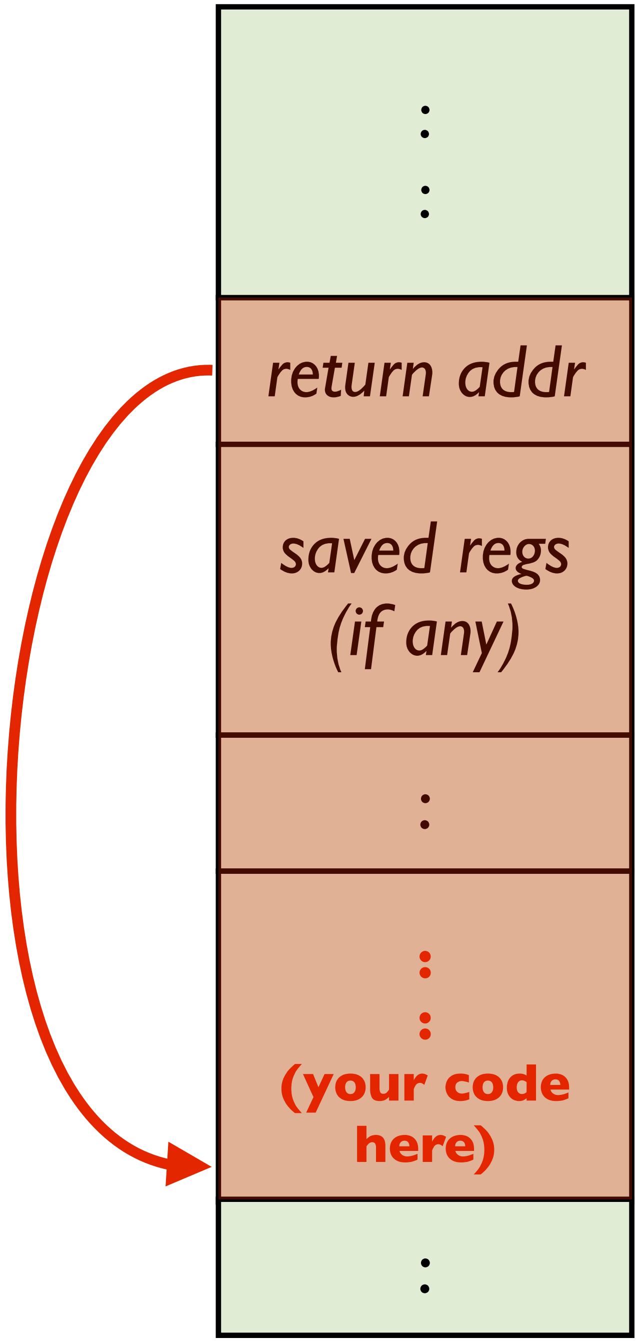
```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

How?

1. overwrite the return address
2. jump *inside the buffer*
3. write x86 code in the buffer

(the write-up tells you which instructions to use)

The Stack in getbuf()



Level 2: Call `bang()`

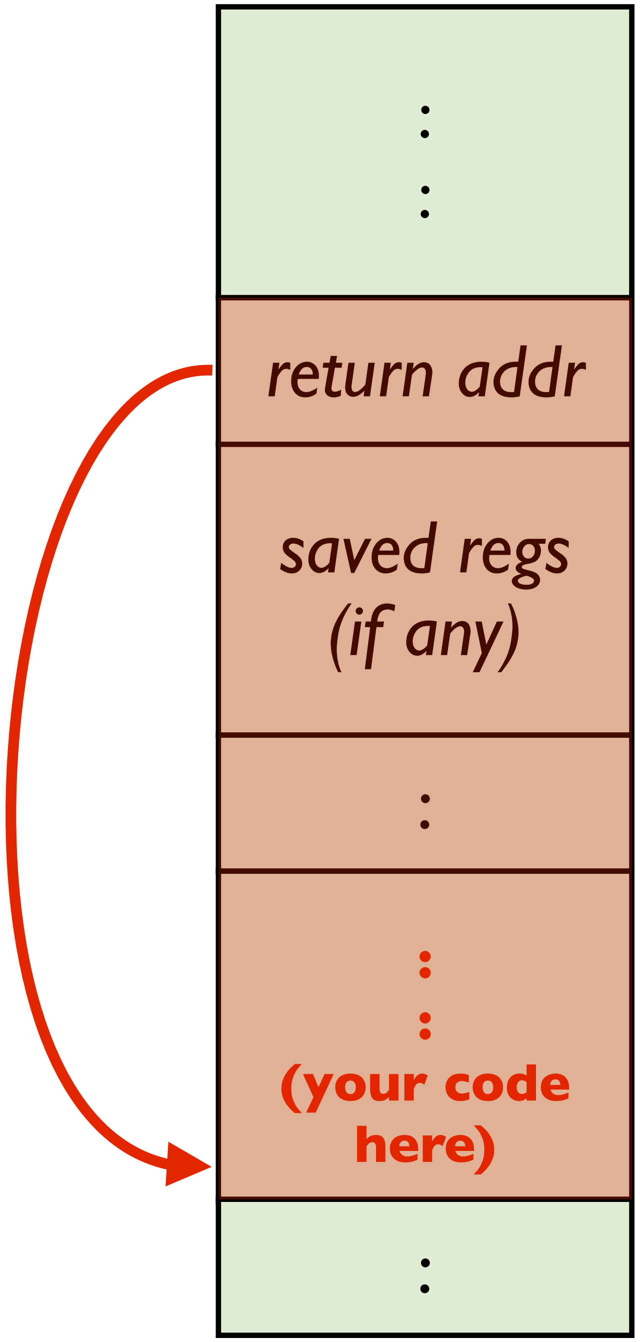
Goal: call bang() after writing your "cookie" to a global variable

```
int getbuf() {
  char buf[36];
  Gets(buf);
  return 1;
}
```

How? Same as before!

1. overwrite the return address
2. jump *inside the buffer*
3. write x86 code in the buffer

The Stack in getbuf()


:
:
return addr
saved regs
(if any)
:
:
:
(your code
here)
: