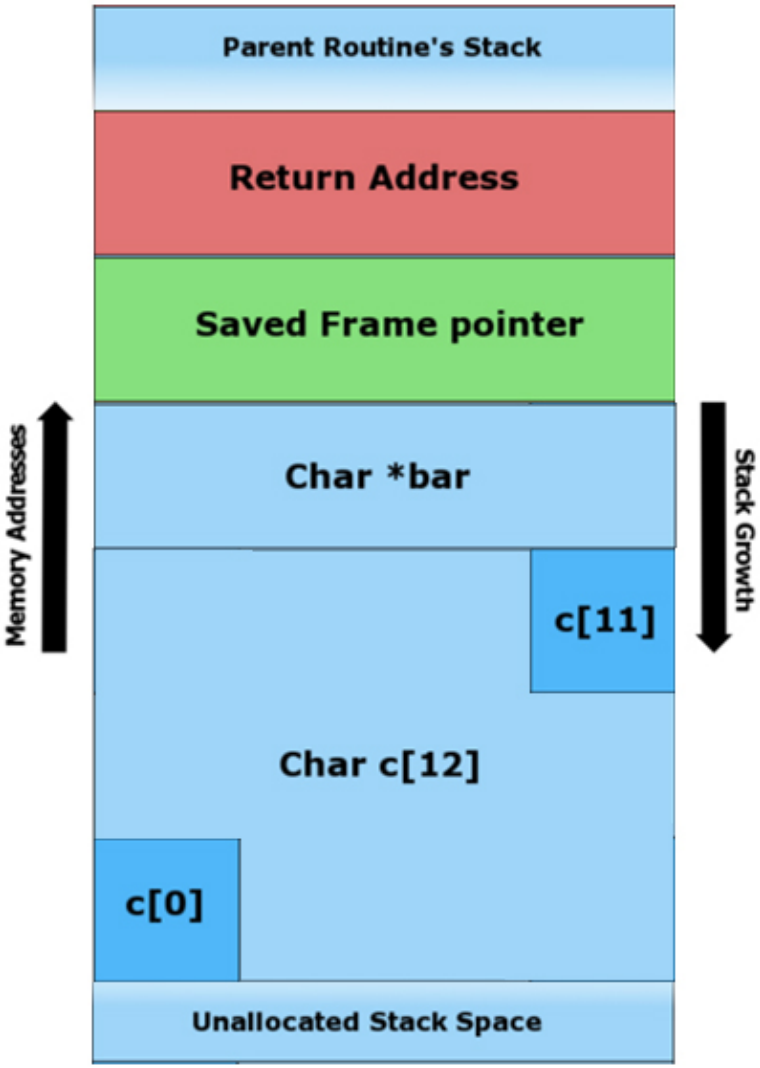
Stack buffer overflow

http://en.wikipedia.org/wiki/Stack_buffer_overflow

What is a stack buffer overflow?

- Caused when a program writes more data to a buffer on the stack than what was initially allocated for the buffer
- Causes bugs, crashes, and can be used in an attack known as stack smashing (executing arbitrary code on a protected machine)
  - Notable Example: Twilight Hack

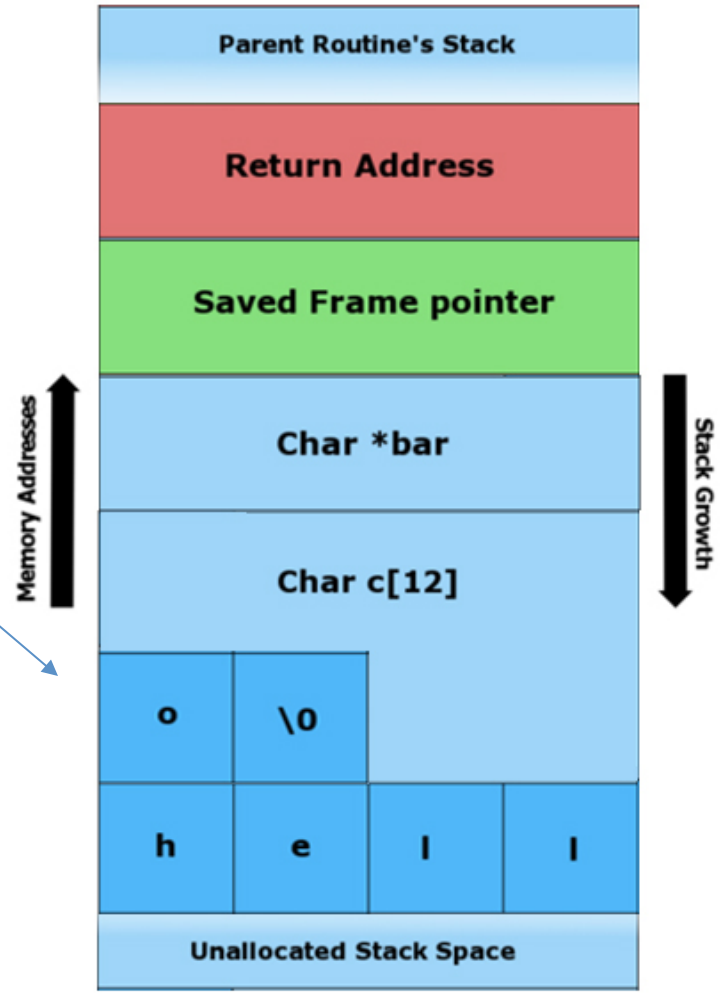
```
#include <string.h>

void foo (char *bar)
{
   char c[12];
   strcpy (c, bar); //no bound
}

int main (int argc, char **argv)
{
   foo(argv[1]);

}
```



Normal Execution

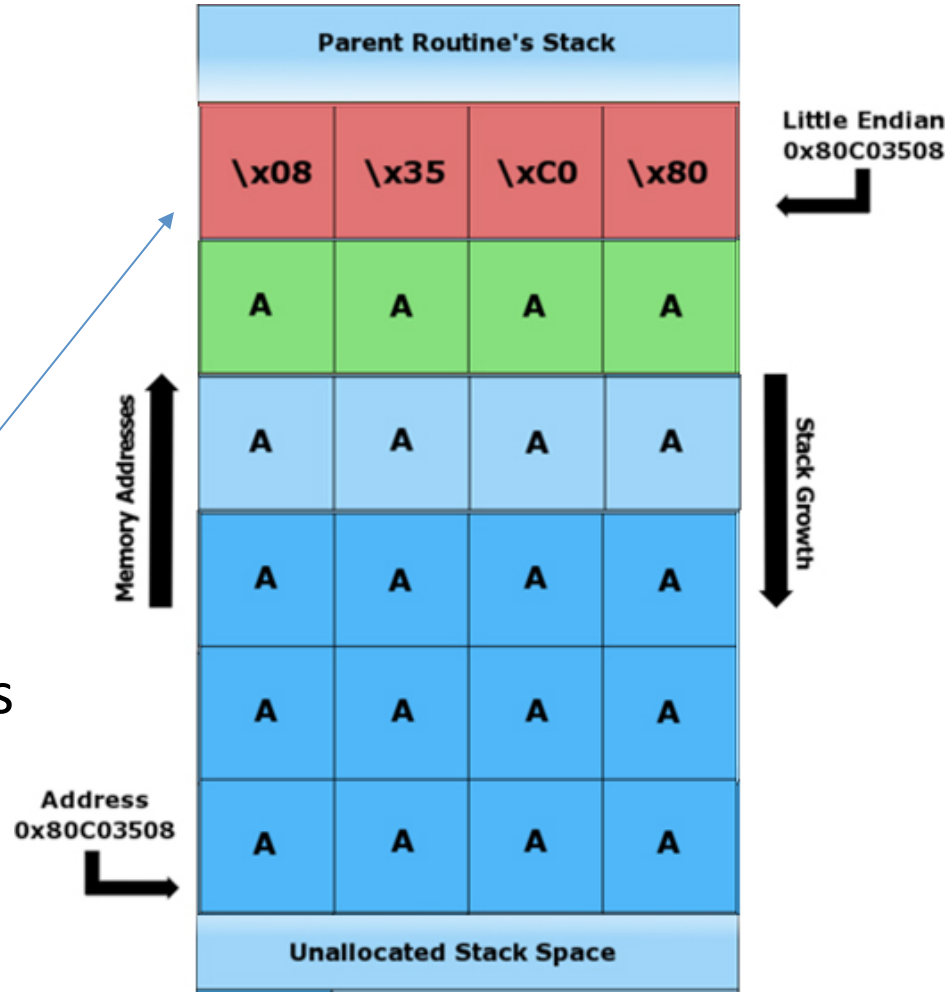
“hello” is written to the char buffer. Note the null terminating byte.

```
#include <string.h>

void foo (char *bar)
{
  char c[12];
  strcpy (c, bar); //no bound
}

int main (int argc, char **argv)
{
  foo(argv[1]);

}
```



Buffer Overflow!

Called with argument:
AAAAAAAAAAAAAAAAAAAA
\x08\x35\xC0\x80



The return address now points to the start of the 12-byte buffer.

Lab 3

- Due Wednesday April 15th.
- Stack buffer overflow problem, very similar to what we have described today.

Tools and Examples