CSED211: Lab. 4

AttackLab

조승혁
shhj1998@postech.ac.kr

POSTECH

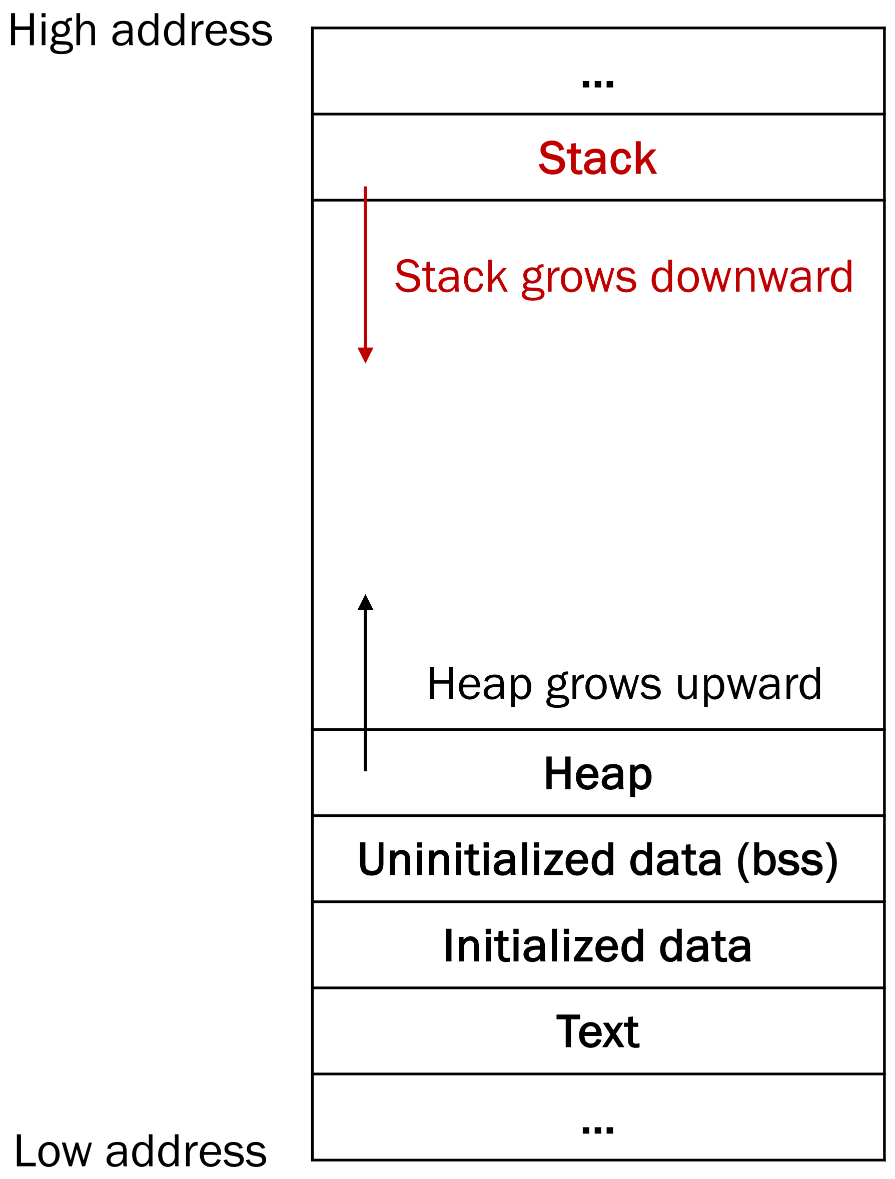
2023.10.16

Table of Contents

- Memory Layout (cont.)
- Buffer Overflow
- Homework: Attack Lab

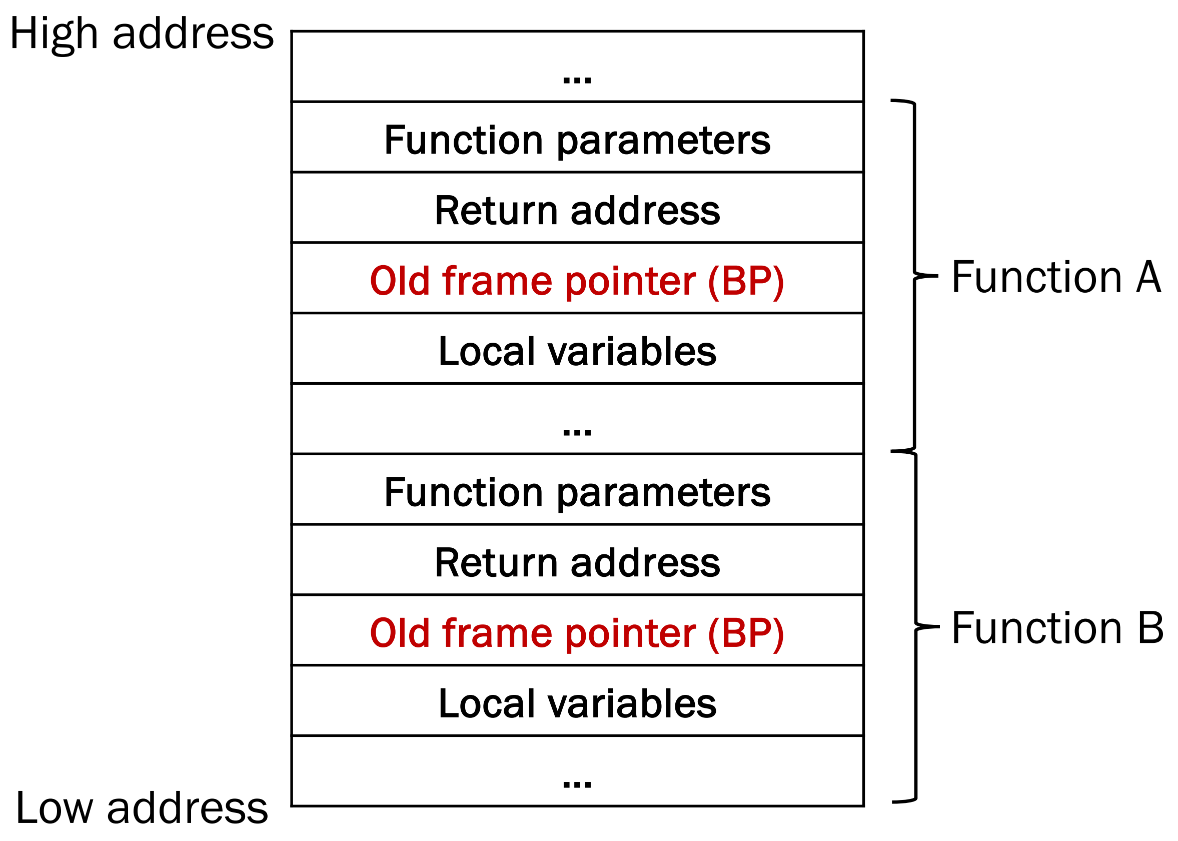
Recap: Memory Layout

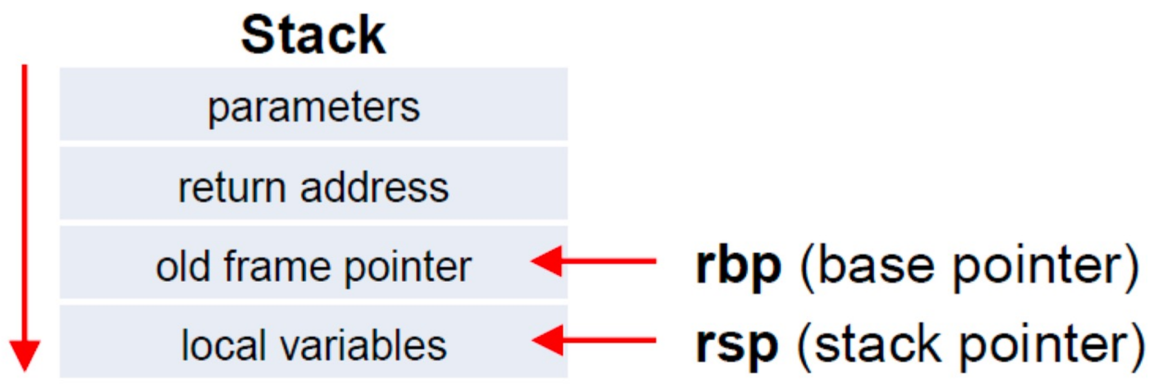
- Each program has its own address space.
  - Stack is Last-In-First-Out (LIFO) data structure.
  - Size of stack varies as the program process.
  - Local variables are stored in stack.



Recap: Stack Frame

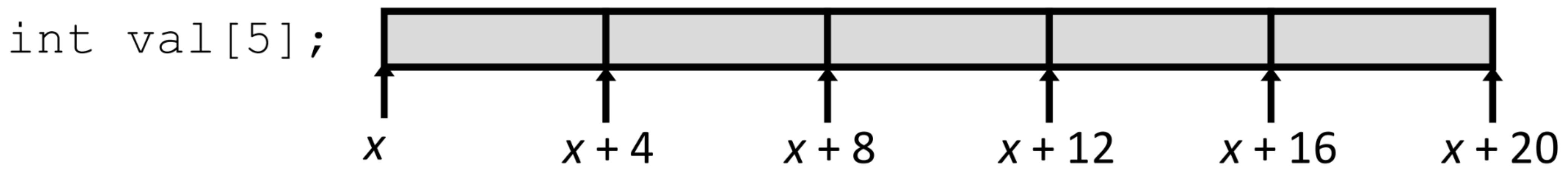
- Stack stores information about active subroutines of a computer program.





Array

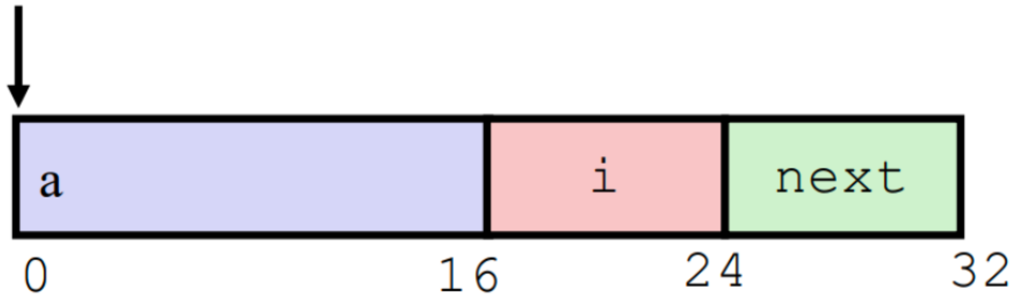
- T A[L];
  - Array of data type T and length L.
  - Address of A[idx] = A + idx * K, where K is the size of T.



Structure

- Elements of structure are ordered and aligned in memory.
  - Address of r.a[idx] = r + 4 * idx.
  - Address of r.i = r + 16, where 16 is the total size of r.a.

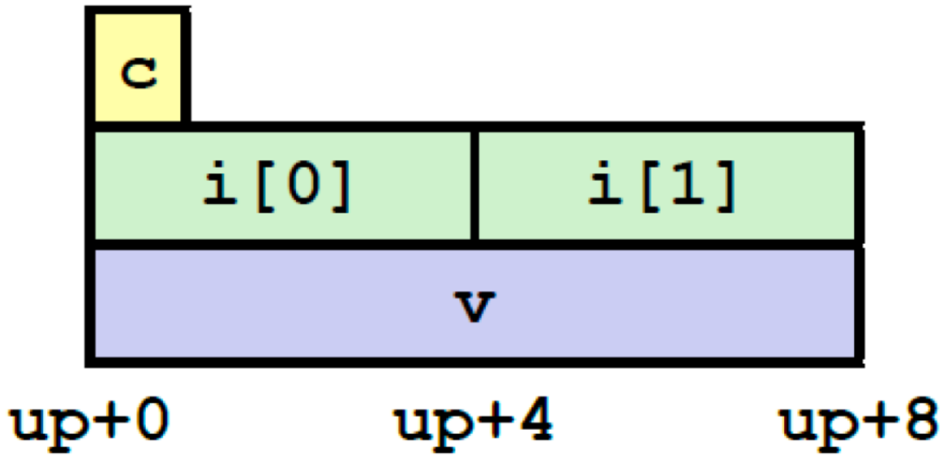
```
struct rec {
    int a[4];
    size_t i;
    struct rec *next;
};
```



Union

- Elements of union are in the same memory.
  - Address of U1.c = U1
  - Address of U1.i[idx] = U1 + 4 * idx.

```
union U1 {
  char c;
  int i[2];
  double v;
}     ;
```

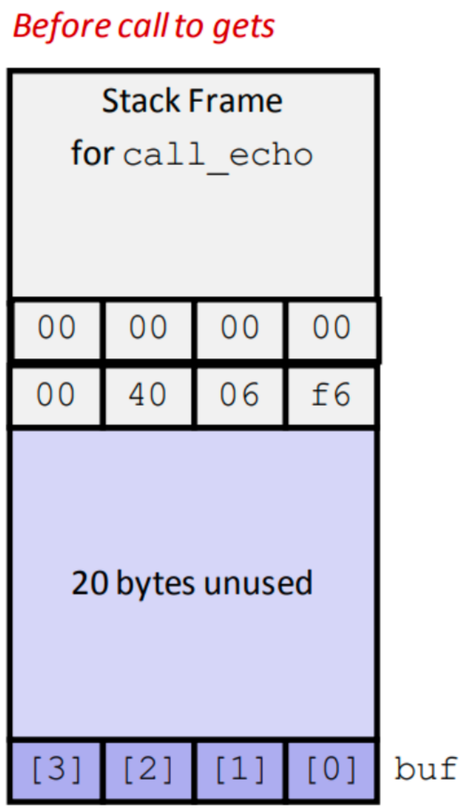


Buffer Overflow (1)

- When the function `echo` is called, the memory layout becomes the right figure.
  - Q) How can we corrupt the other part with `gets`?

```
void echo()
{
    char buf[4];
    gets(buf);
    . . .
}
```

```
echo:
  subq   $24, %rsp
  movq   %rsp, %rdi
  call   gets
  . . .
```



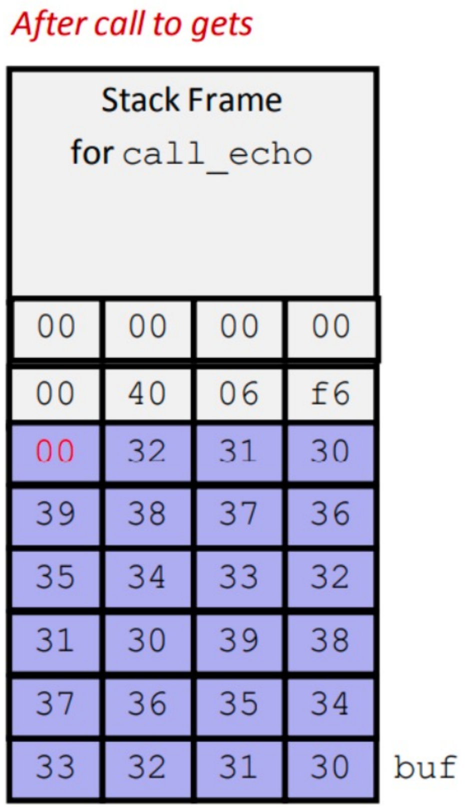
Buffer Overflow (2)

- A) If we input a string longer than 4 bytes (the size of `buf`), the other parts are corrupted.
  - By inserting different address, we can call different function in the program.

```
void echo()
{
    char buf[4];
    gets(buf);
    . . .
}
```

```
echo:
  subq  $24, %rsp
  movq  %rsp, %rdi
  call  gets
  . . .
```

```
unix>./bufdemo
Type a string:01234567890123456789012
01234567890123456789012
```

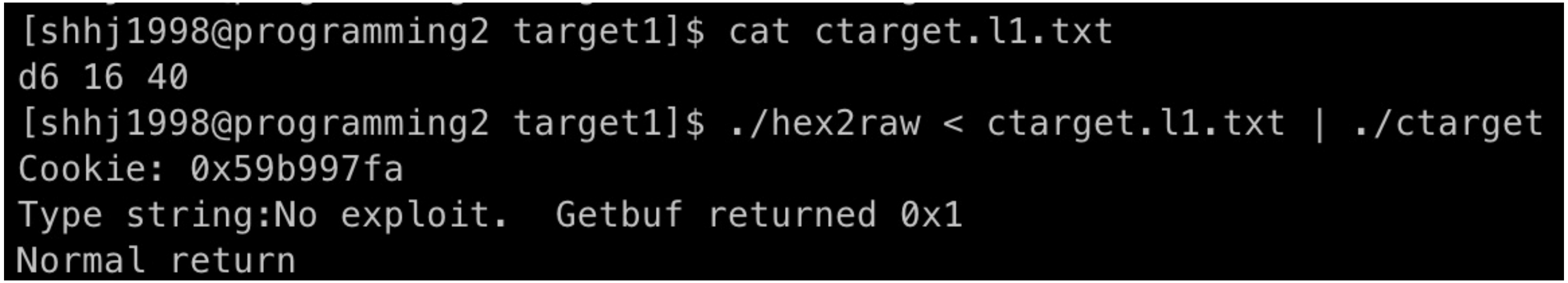


Homework (Attack Lab)

- Make sure that you enable local forwarding to access attack server.
- To download your target, go to http://127.0.0.1:15513.
  - Enter your information, student ID and school email.
  - Upload your target#.tar to the programming server.
- Your goal is to exploit the **five** targets:
  - ctarget.l1, ctarget.l2, ctarget.l3, rtarget.l2, rtarget.l3.
- Your score (corresponds to target #) will be automatically uploaded at:
  - http://127.0.0.1:15513/scoreboard.
  - Target can be exploited only in the programming server.
  - The score is not updated if you work in other machines.

Homework (Attack Lab): Hex2raw

- We can easily convert a hex string into raw string using `hex2raw`.
  - First, prepare the exploit string in hex format (e.g., 0x4016d6 => d6 16 40).
  - Then, run the program with the hex string as follow:



```
[shhj1998@programming2 target1]$ cat ctarget.l1.txt
d6 16 40
[shhj1998@programming2 target1]$ ./hex2raw < ctarget.l1.txt | ./ctarget
Cookie: 0x59b997fa
Type string:No exploit.  Getbuf returned 0x1
Normal return
```

Homework (Attack Lab)

- You can find more details in `writeup_attacklab.pdf`.

Homework

- Deadline: 11/6 23:59 (Mon)
- You need to upload a report in the PLMS.
  - Explain how did you exploit the target programs in the report.
  - Follow the file name format, [student#].pdf.
    - For example, 2020xxxx.pdf (No square brackets in the file name).
    - No doc, No zip!