Access Control and Privacy Policies (10)

Email: christian.urban at kcl.ac.uk
Office: S1.27 (1st floor Strand Building)
Slides: KEATS (also homework is there)

Revision

1st Lecture

2nd Lecture: E-Voting

- Integrity
- Ballot Secrecy
- Voter Authentication
- Enfranchisement
- Availability

2nd Lecture: E-Voting

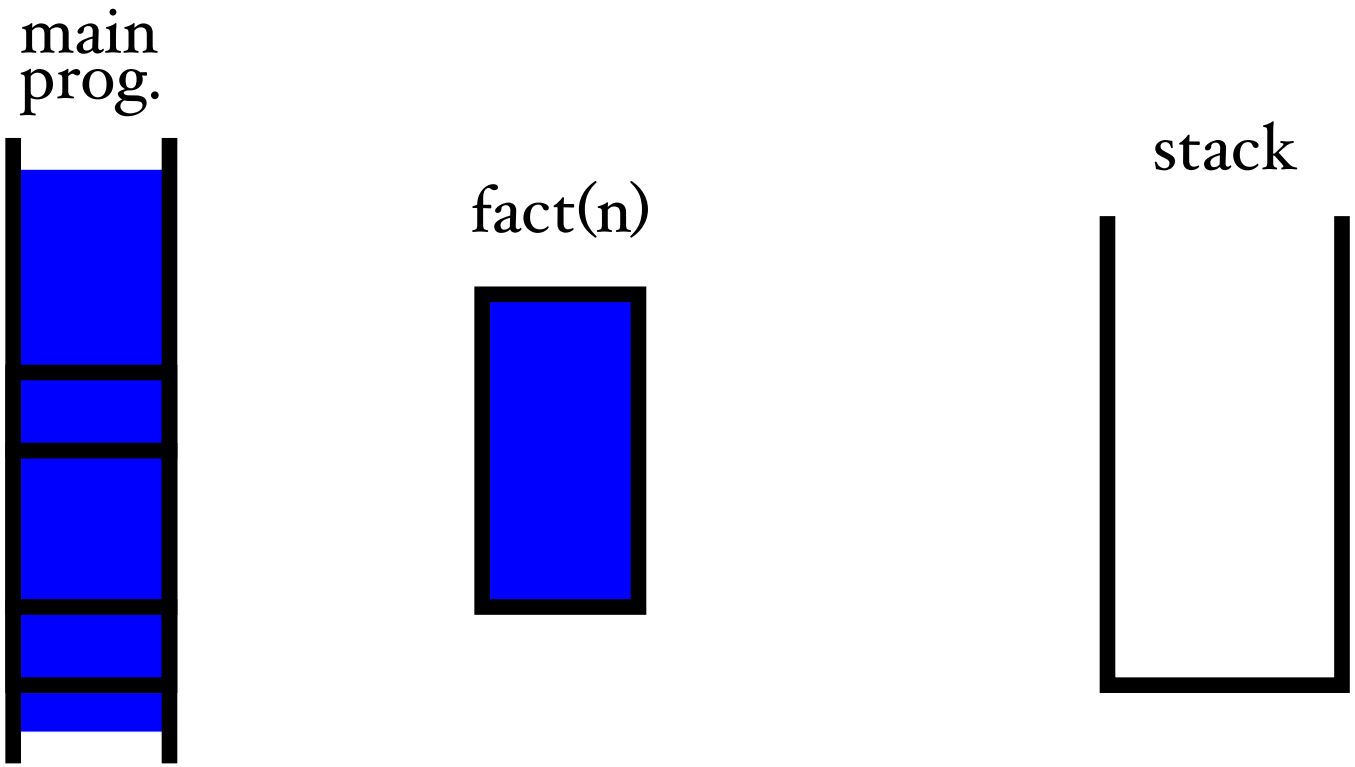
Online Banking vs. E-Voting

- online banking: if fraud occurred you try to identify who did what (somebody's account got zero)

- e-voting: some parts can be done electronically, but not the actual voting (final year project: online voting)

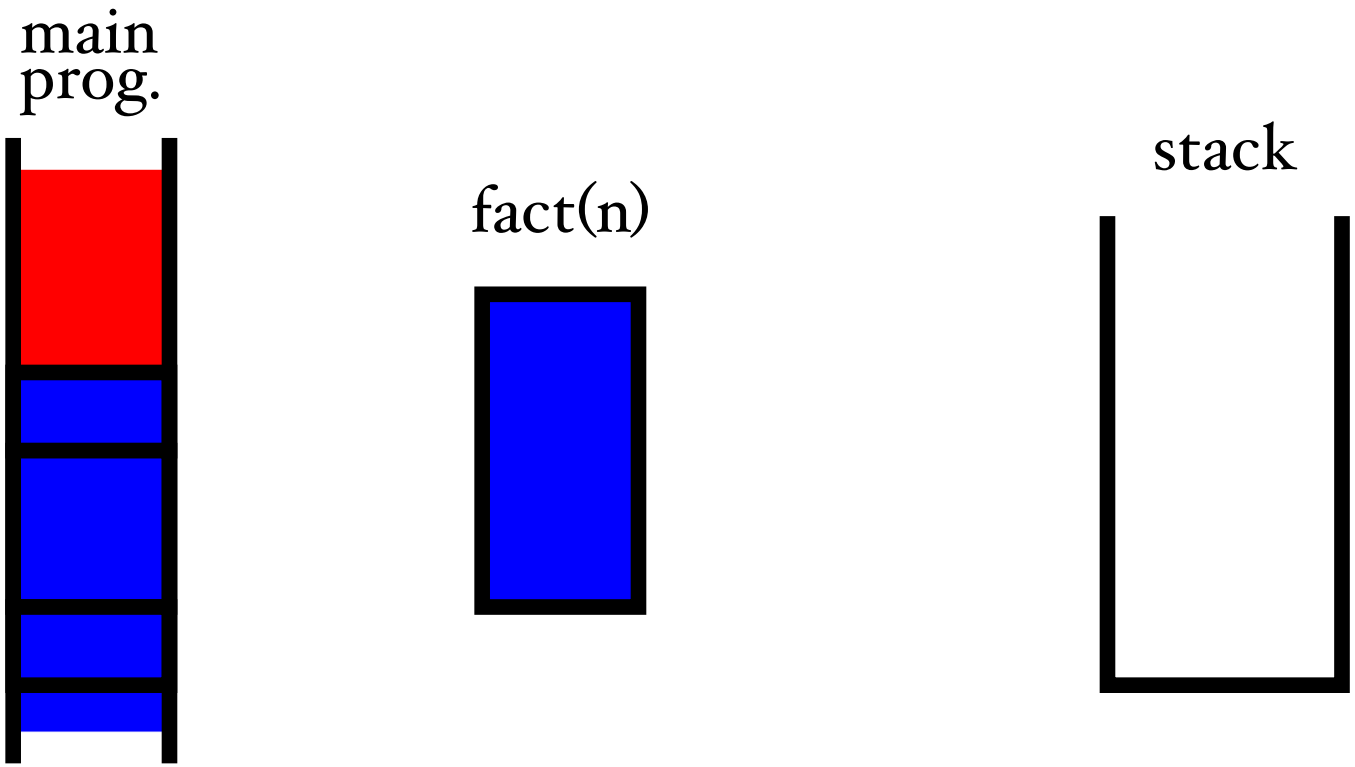
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



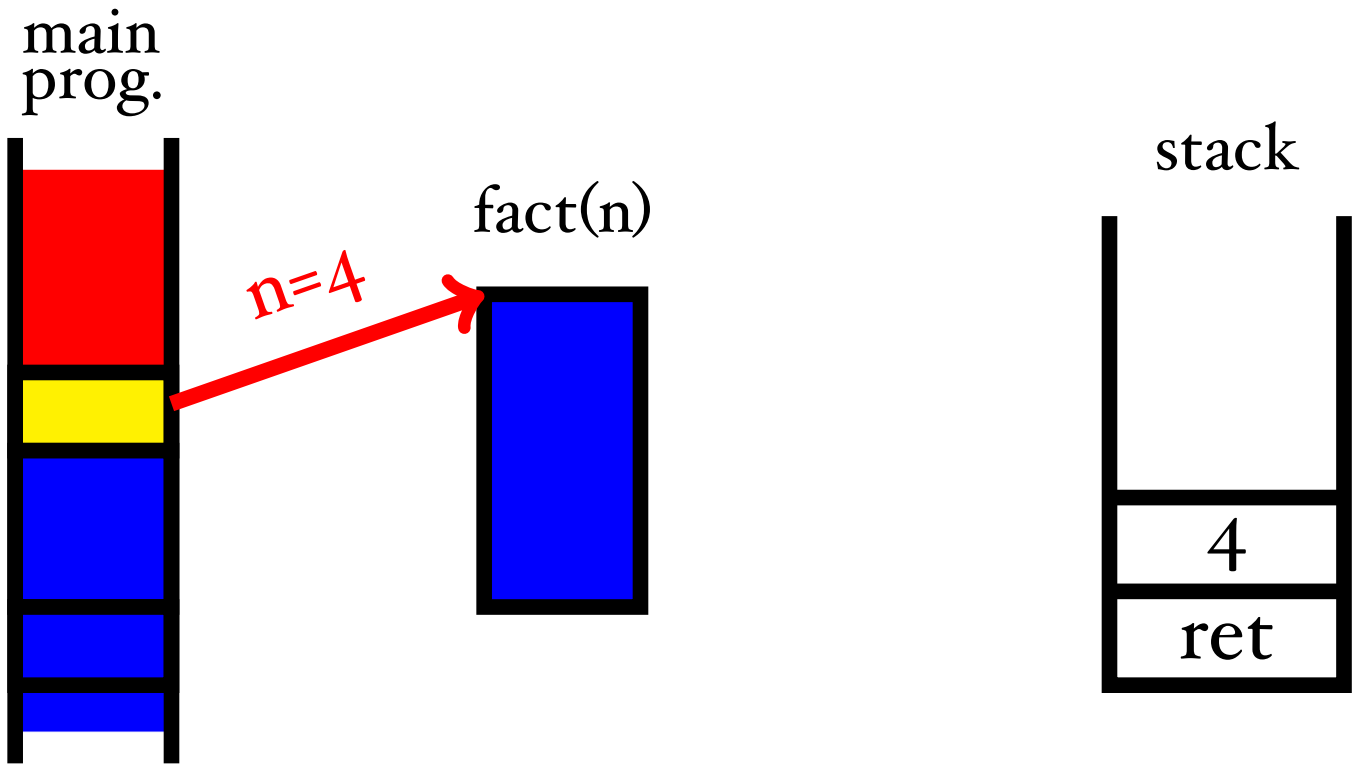
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



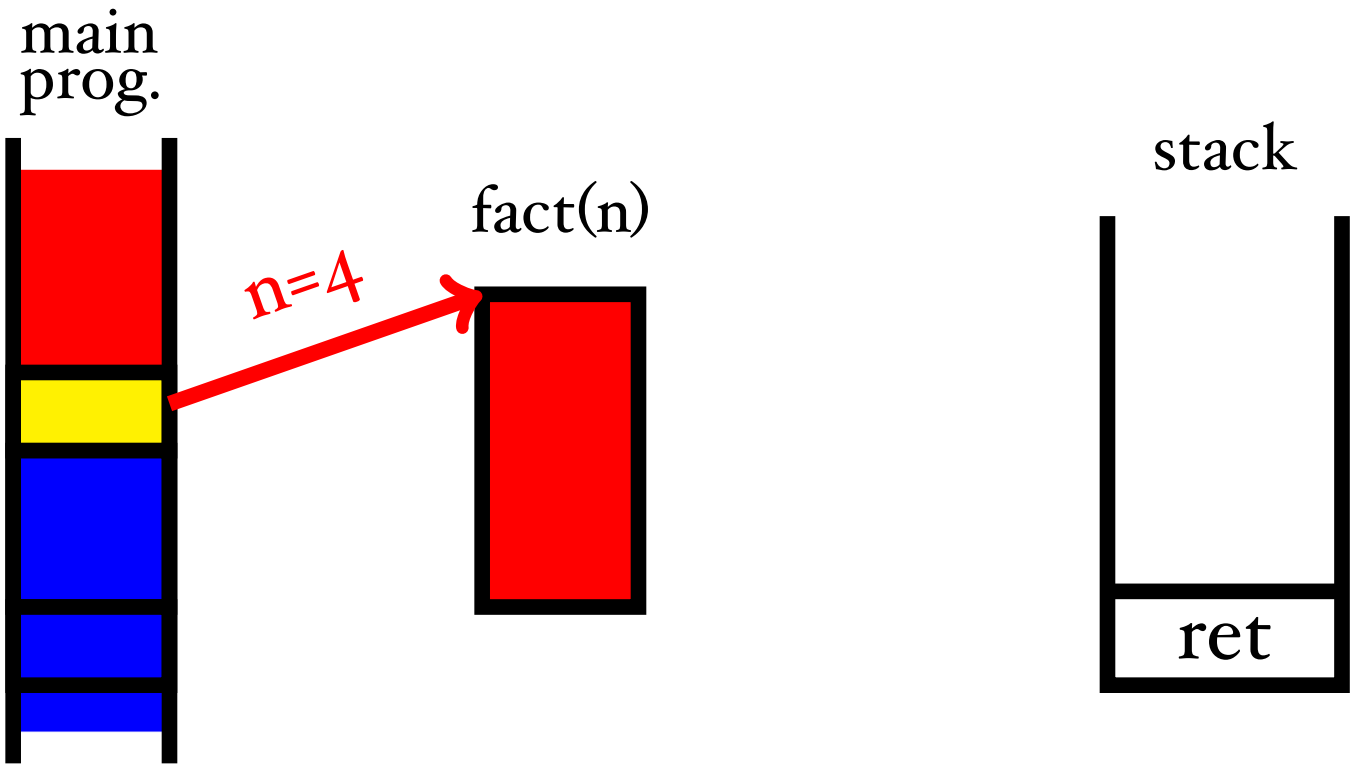
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



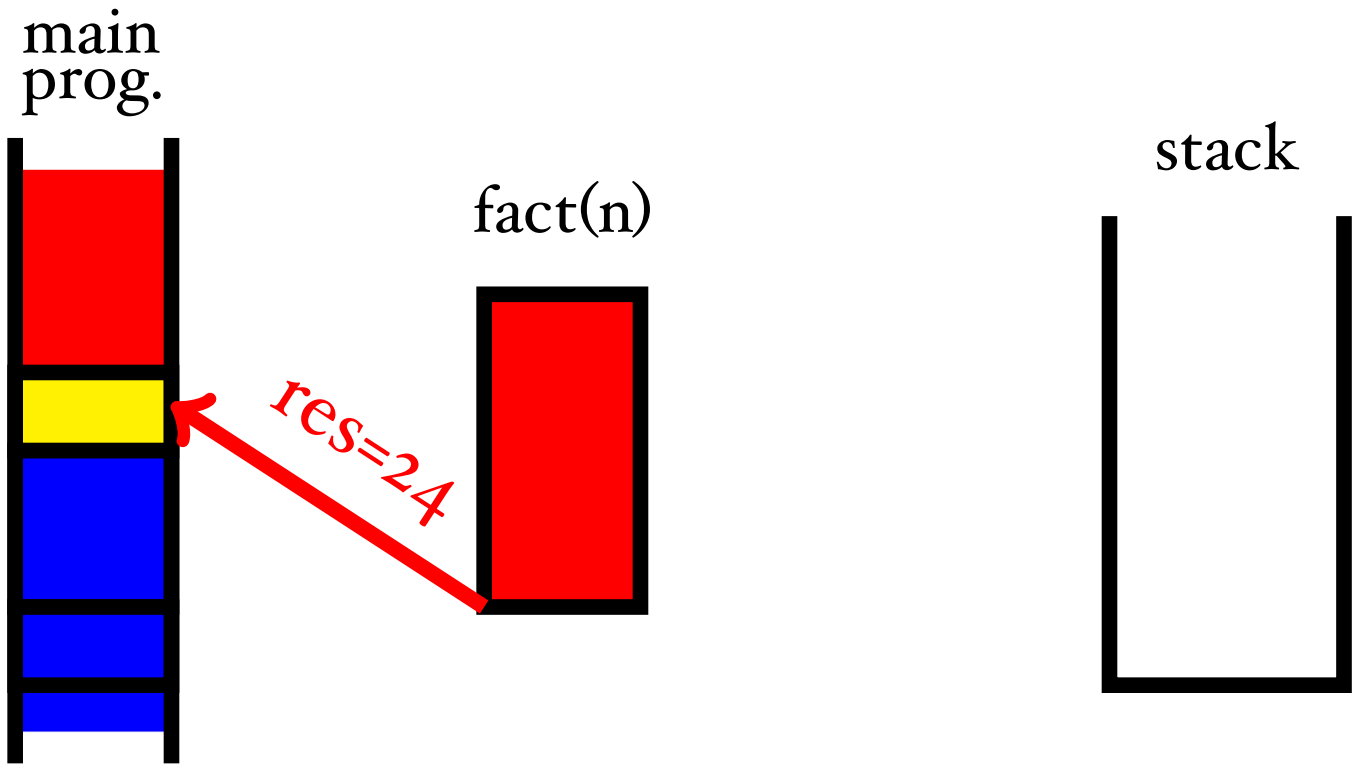
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



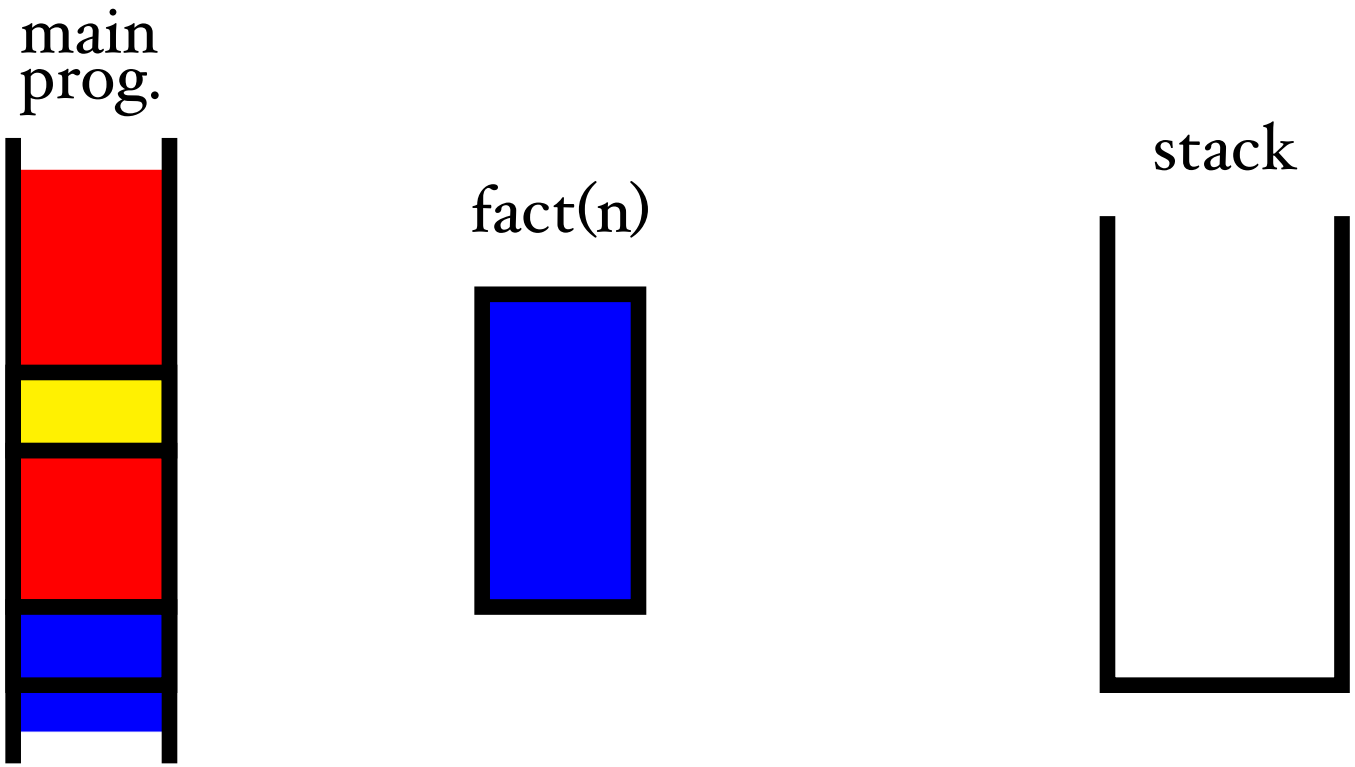
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



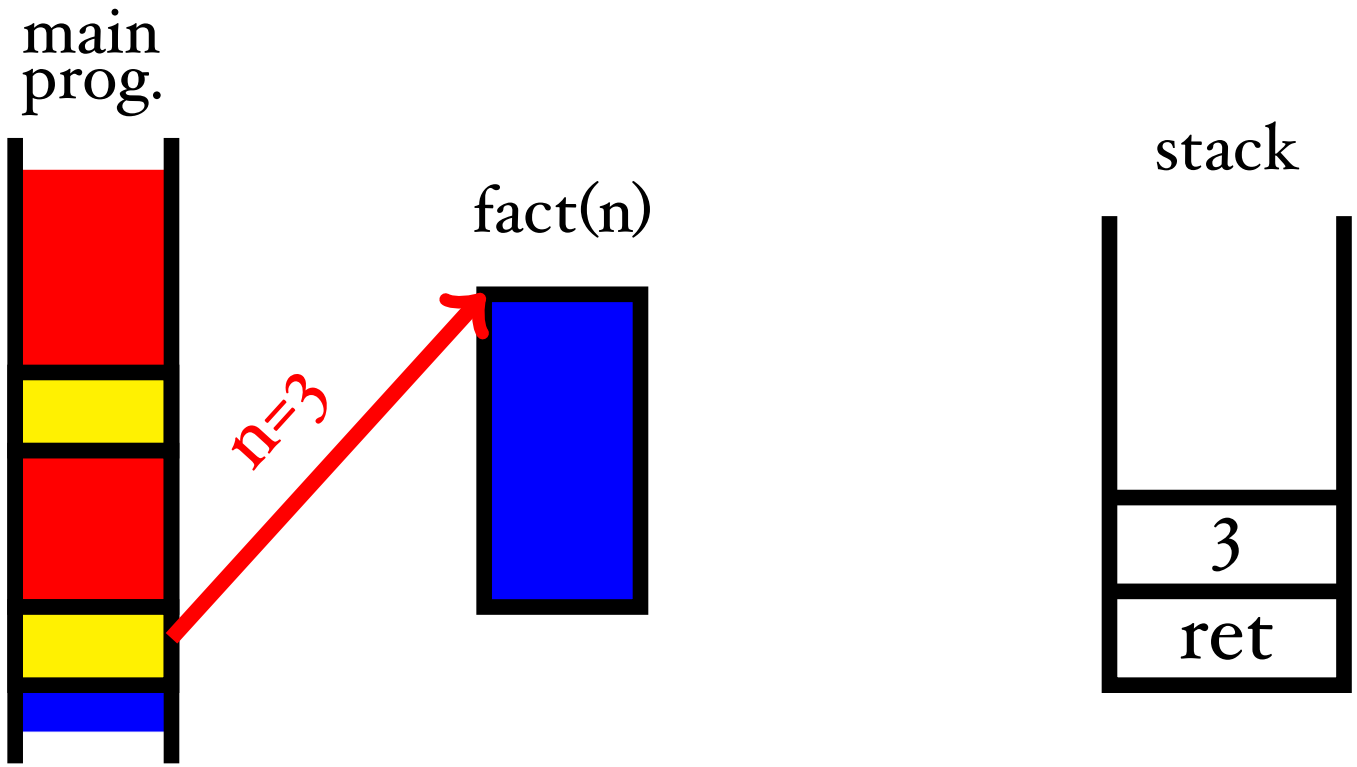
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



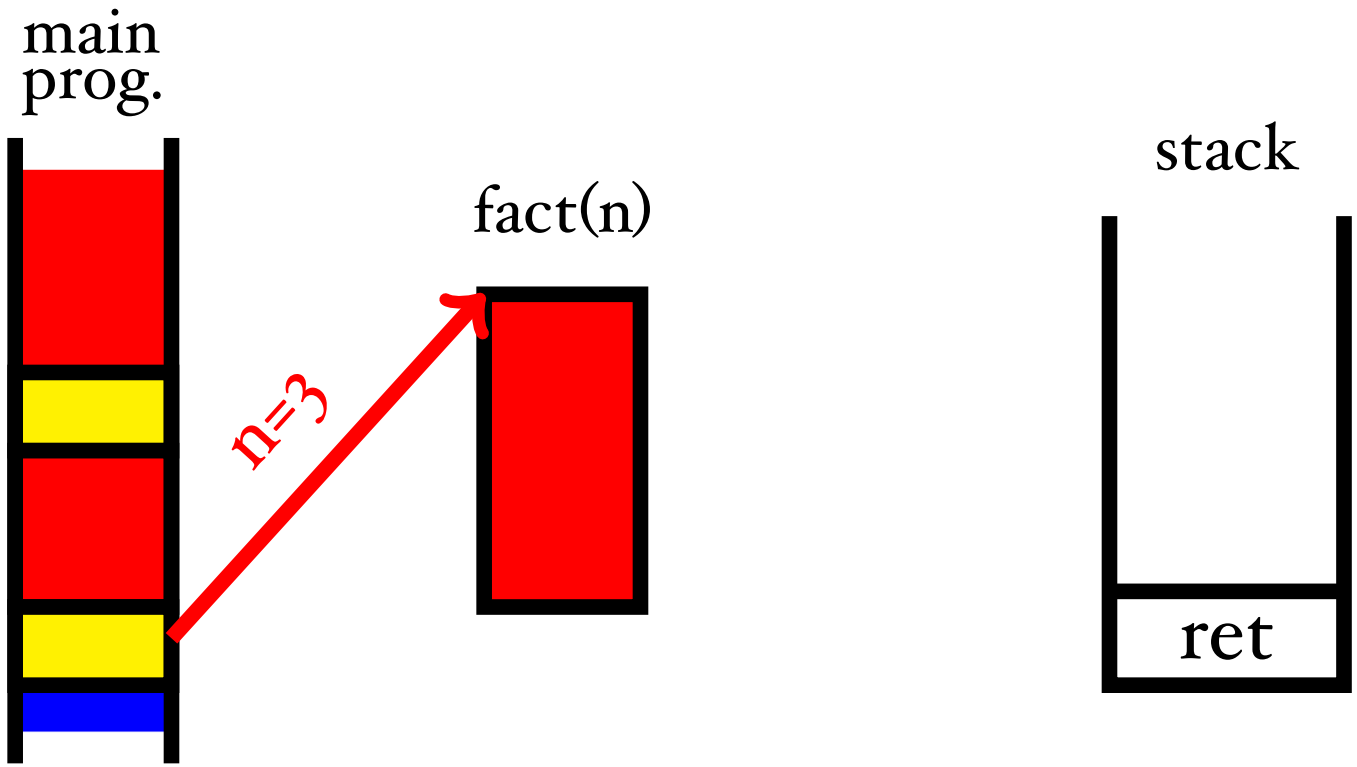
3rd Lecture: Buffer Overflow Attacks

- the problem arises from the way C/C++ organises its function calls



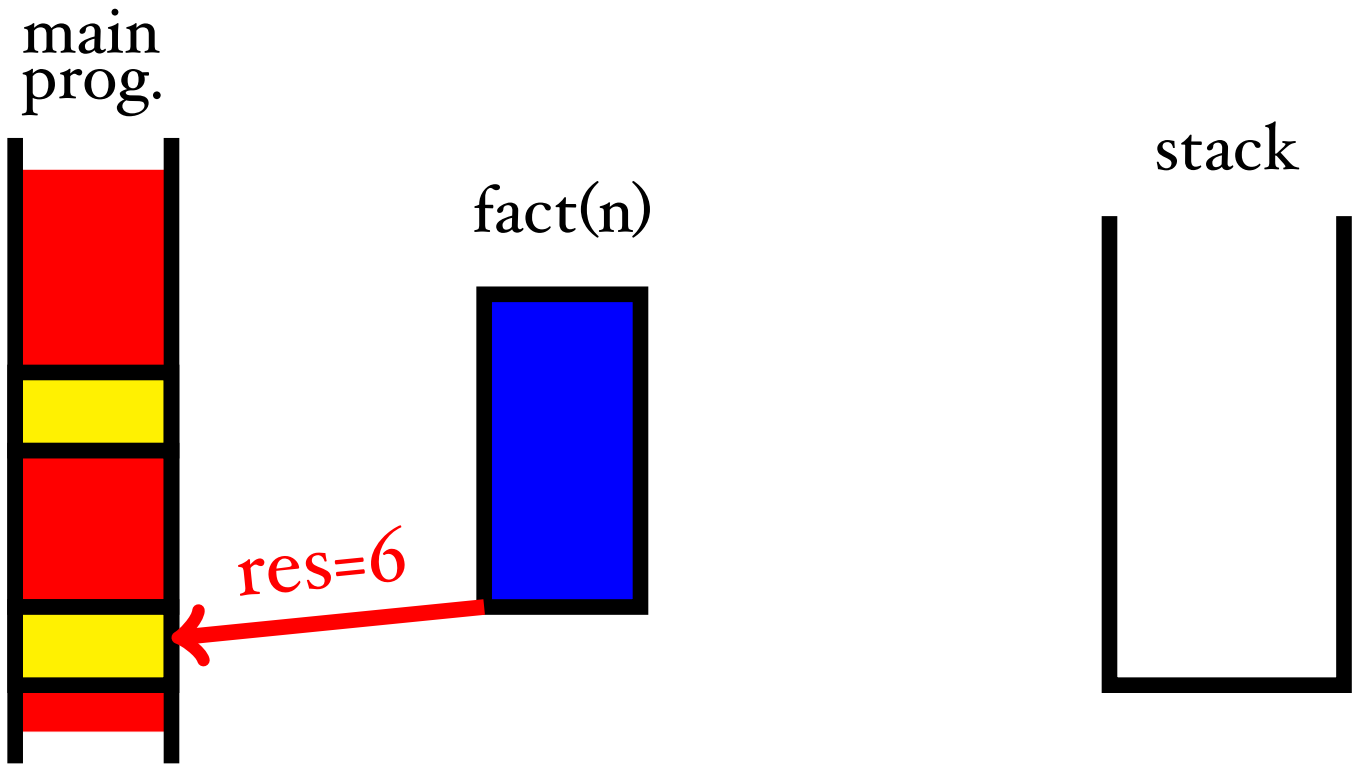
3rd Lecture: Buffer Overflow Attacks

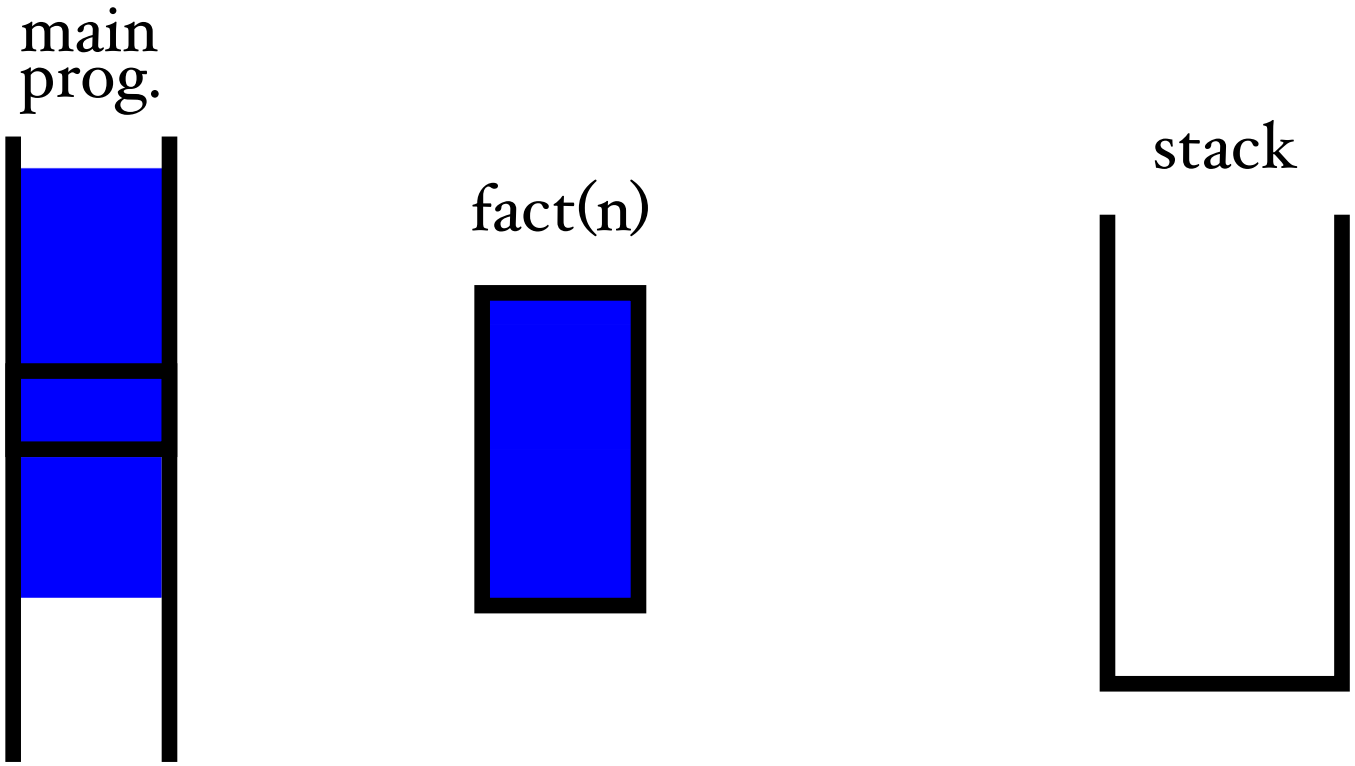
- the problem arises from the way C/C++ organises its function calls

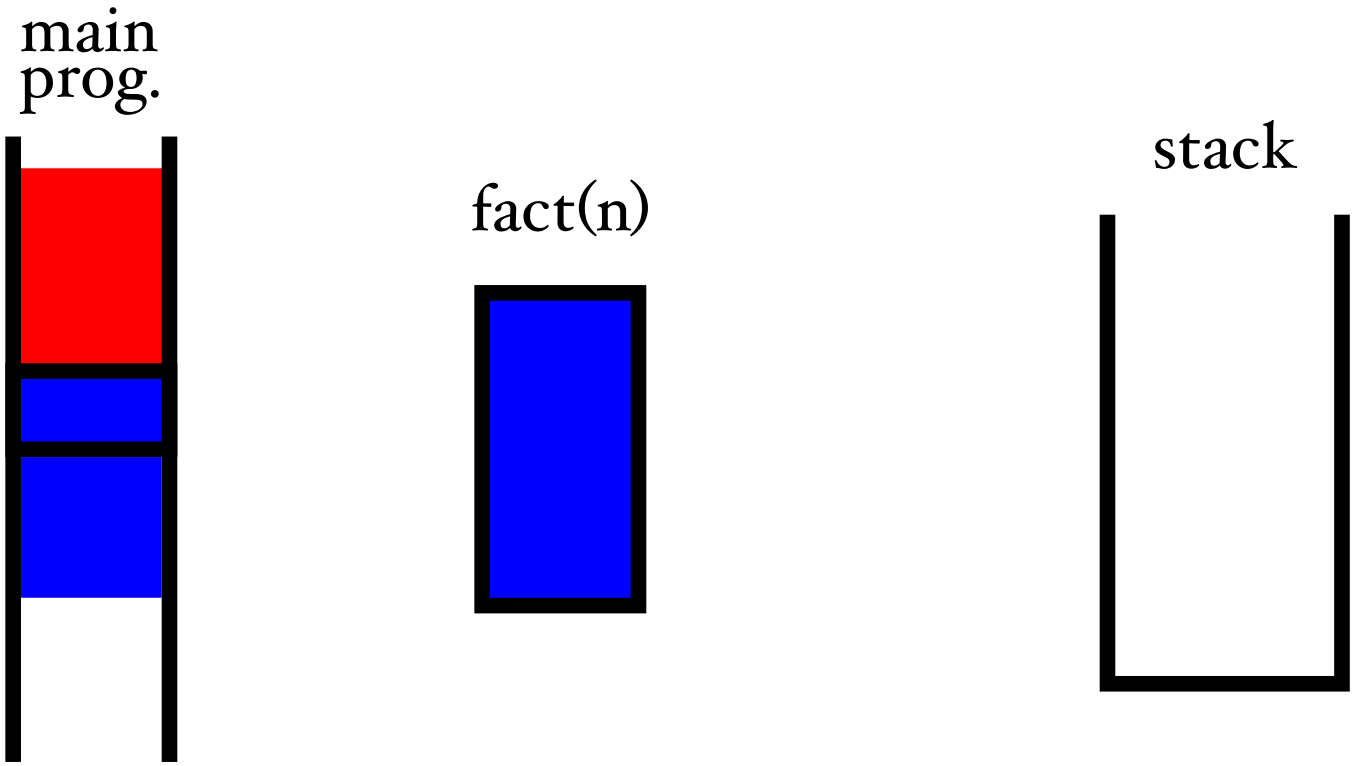


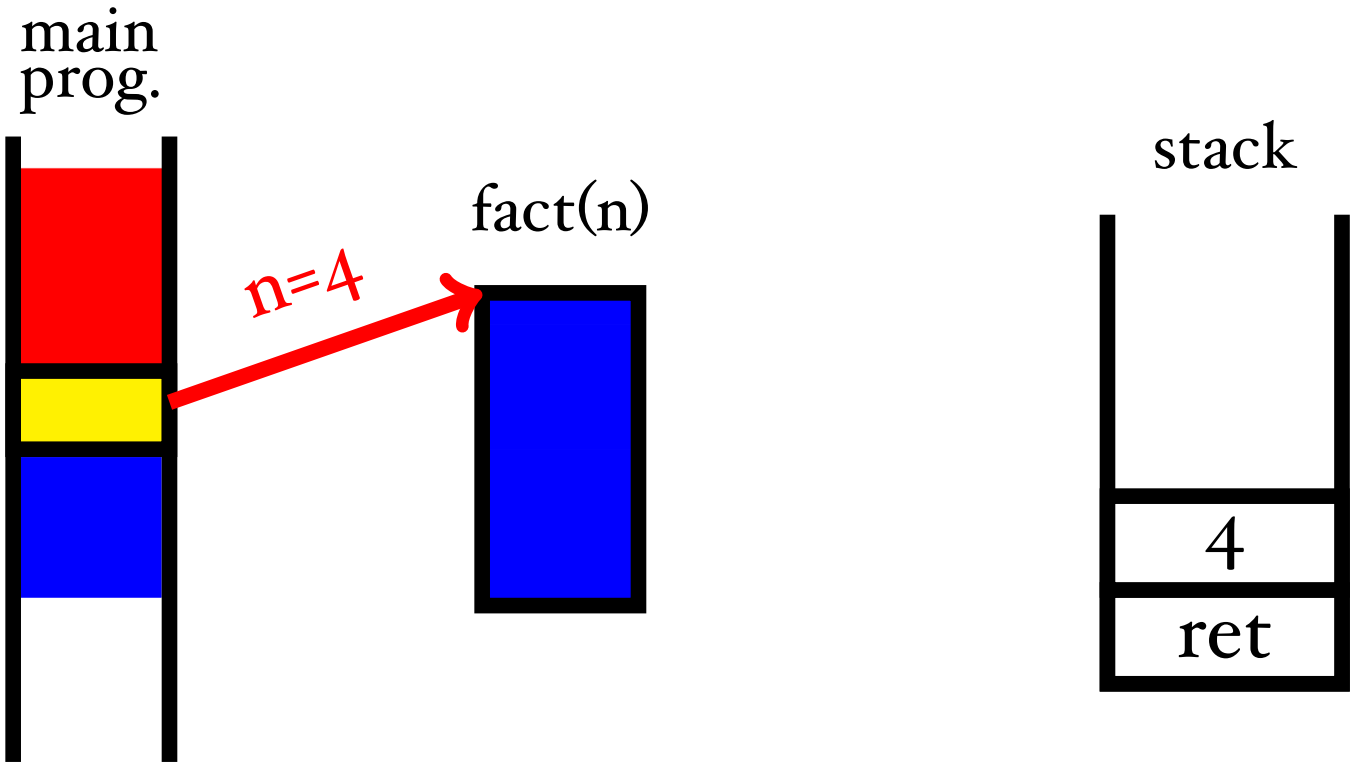
3rd Lecture: Buffer Overflow Attacks

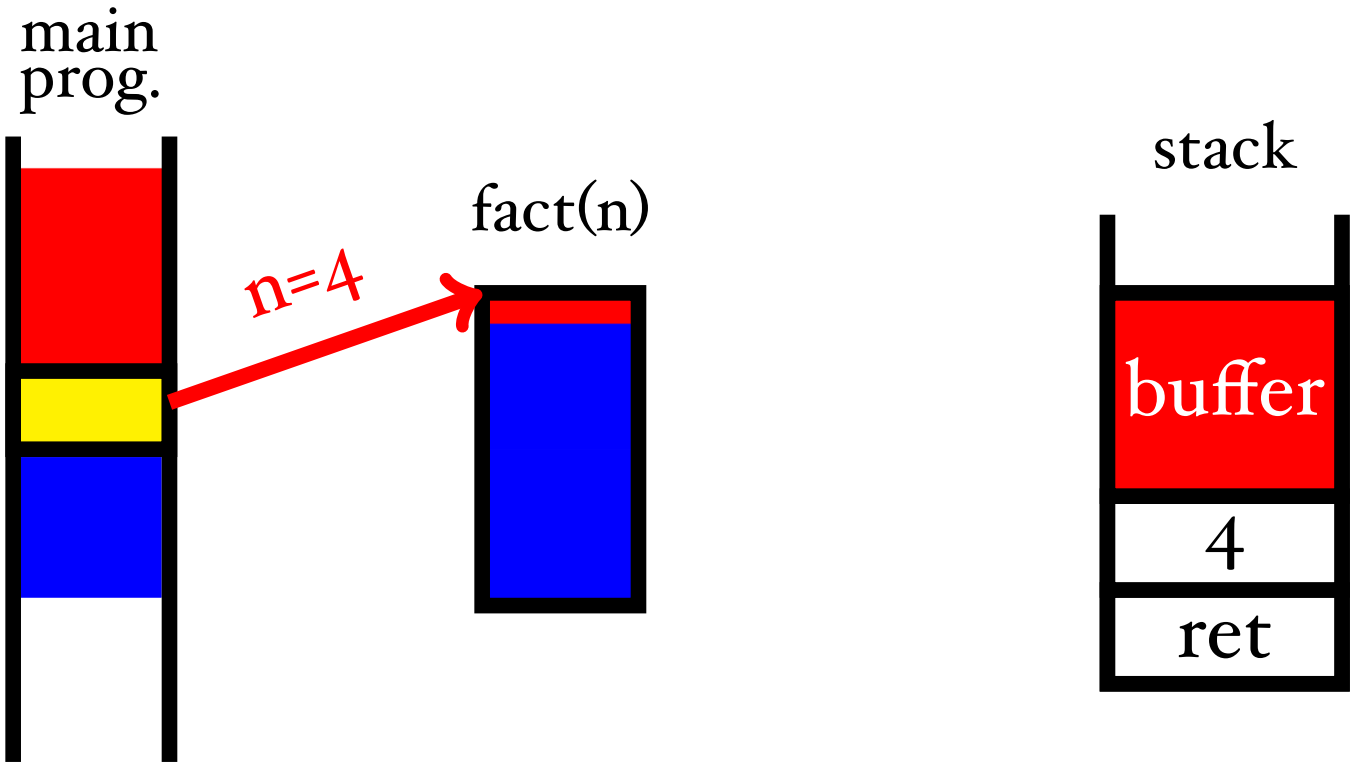
- the problem arises from the way C/C++ organises its function calls

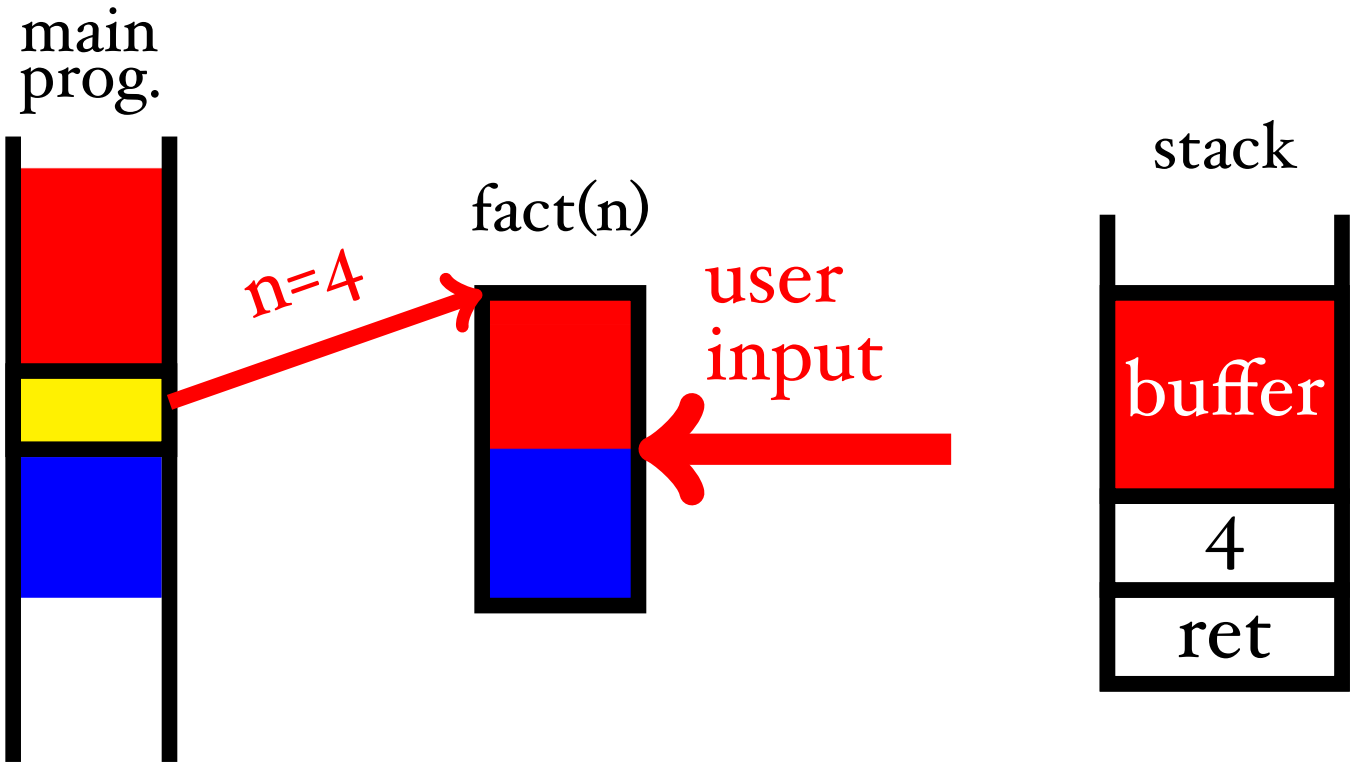


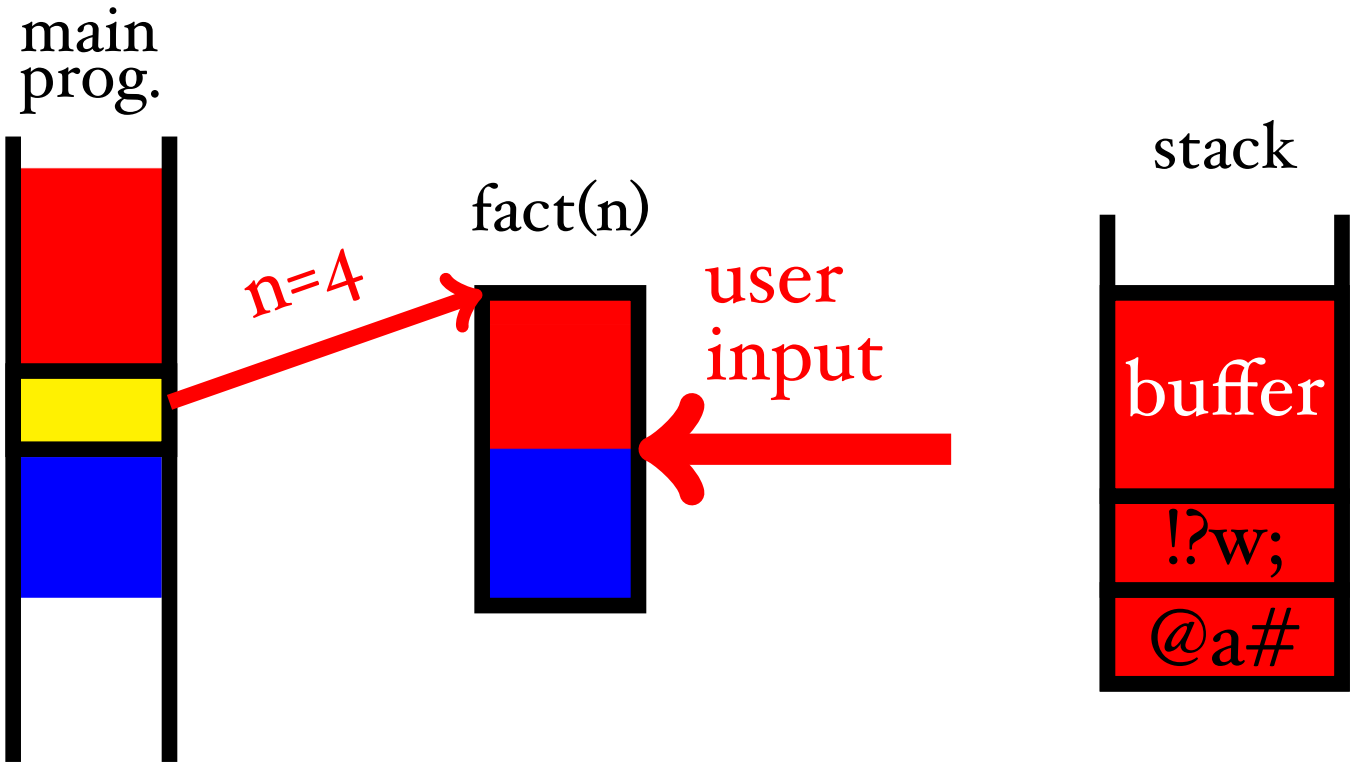

main
prog.
fact(n)
stack

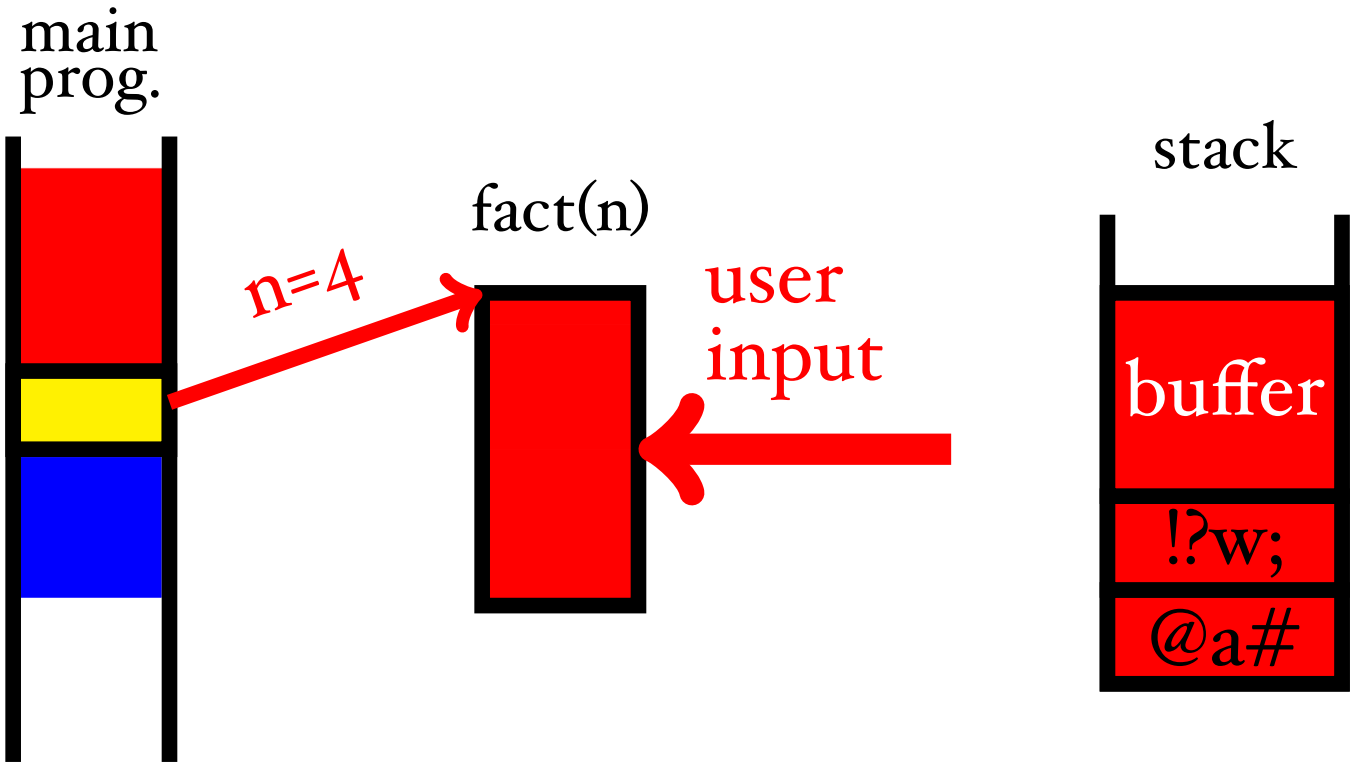

main
prog.
fact(n)
stack

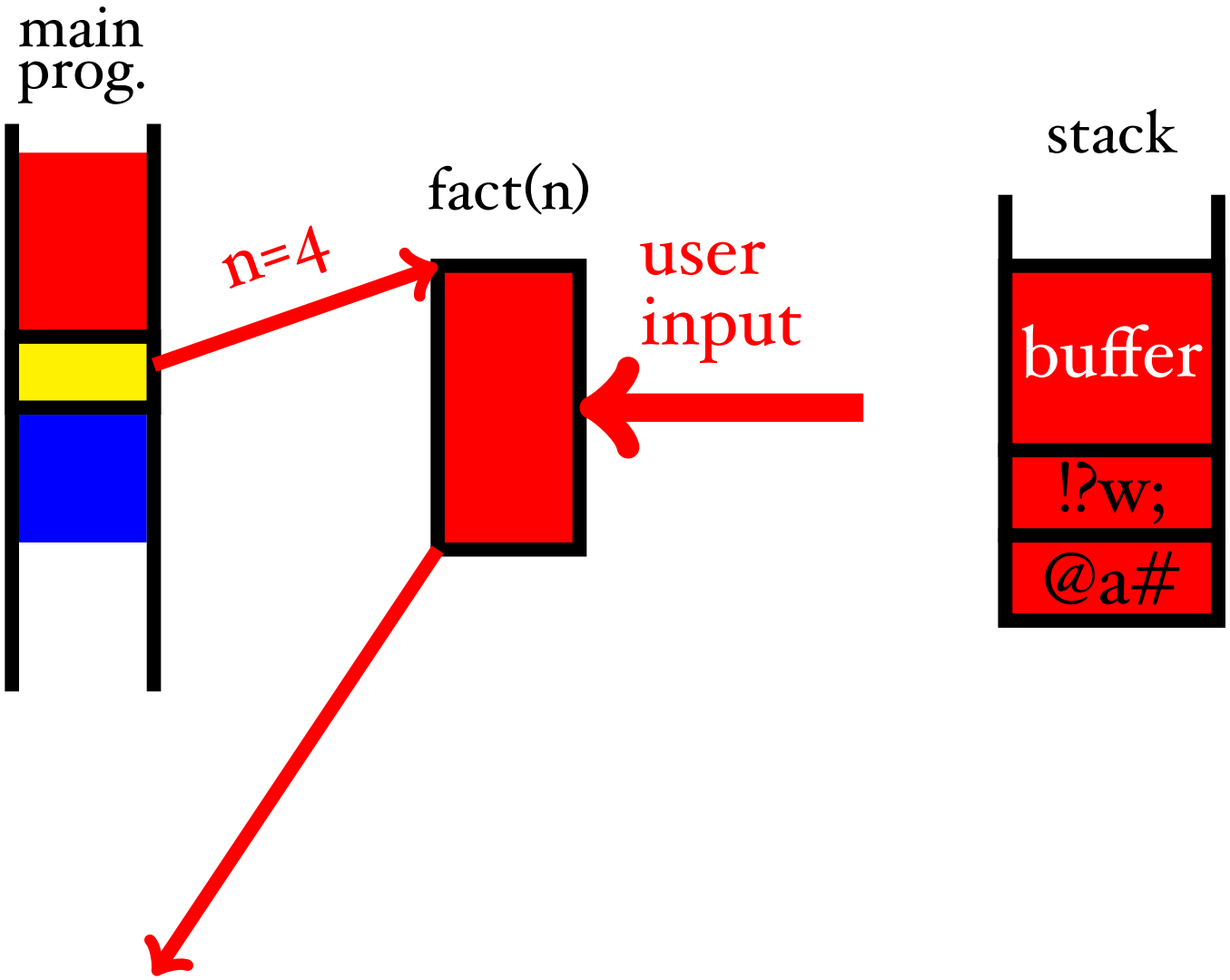

main
prog.
n=4
fact(n)
stack
4
ret


main
prog.
n=4
fact(n)
stack
buffer
4
ret


main
prog.
n=4
fact(n)
user
input
stack
buffer
4
ret

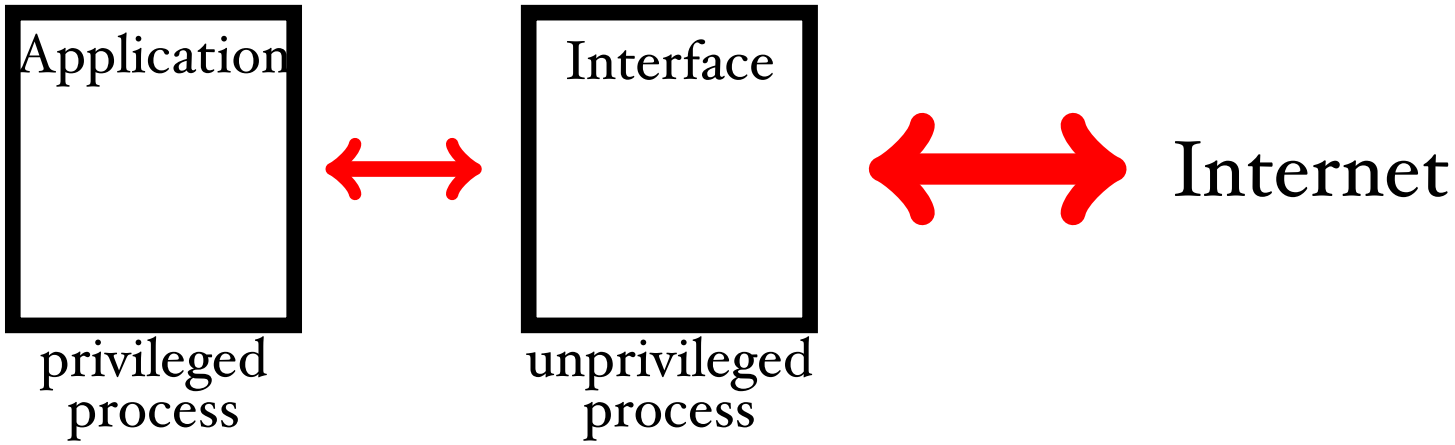

main
prog.
n=4
fact(n)
user
input
stack
buffer
!?w;
@a#


main
prog.
n=4
fact(n)
user
input
stack
buffer
!?w;
@a#


main
prog.
n=4
fact(n)
user
input
stack
buffer
!?w;
@a#

3rd Lecture: Unix Access Control

- privileges are specified by file access permissions ("everything is a file")



- the idea is make the attack surface smaller and mitigate the consequences of an attack

3rd Lecture: Unix Access Control

- when a file with setuid is executed, the resulting process will assume the UID given to the owner of the file

```
$ ls -ld . * */*
drwxr-xr-x 1 ping staff  32768 Apr  2 2010 .
-rw----r-- 1 ping students  31359 Jul 24 2011 manual.txt
-r--rw--w- 1 bob students   4359 Jul 24 2011 report.txt
-rwsr--r-x 1 bob students 141359 Jun  1 2013 microedit
dr--r-xr-x 1 bob staff  32768 Jul 23 2011 src
-rw-r--r-- 1 bob staff  81359 Feb 28 2012 src/code.c
-r--rw---- 1 emma students   959 Jan 23 2012 src/code.h
```

4th Lecture: Security Levels

Bell-LaPadula access model:

- Read Rule: A principal P can read an object O if and only if P's security level is at least as high as O's.
- Write Rule: A principal P can write an object O if and only if O's security level is at least as high as P's.
- Meta-Rule: All principals in a system should have a sufficiently high security level in order to access an object.

4th Lecture: Security Levels

Biba (data integrity)

- Biba: **'no read down' - 'no write up'**
- Read Rule: A principal P can read an object O if and only if P's security level is lower or equal than O's.
- Write Rule: A principal P can write an object O if and only if O's security level is lower or equal than P's.

4th Lecture: Protocols

A mutual authentication protocol

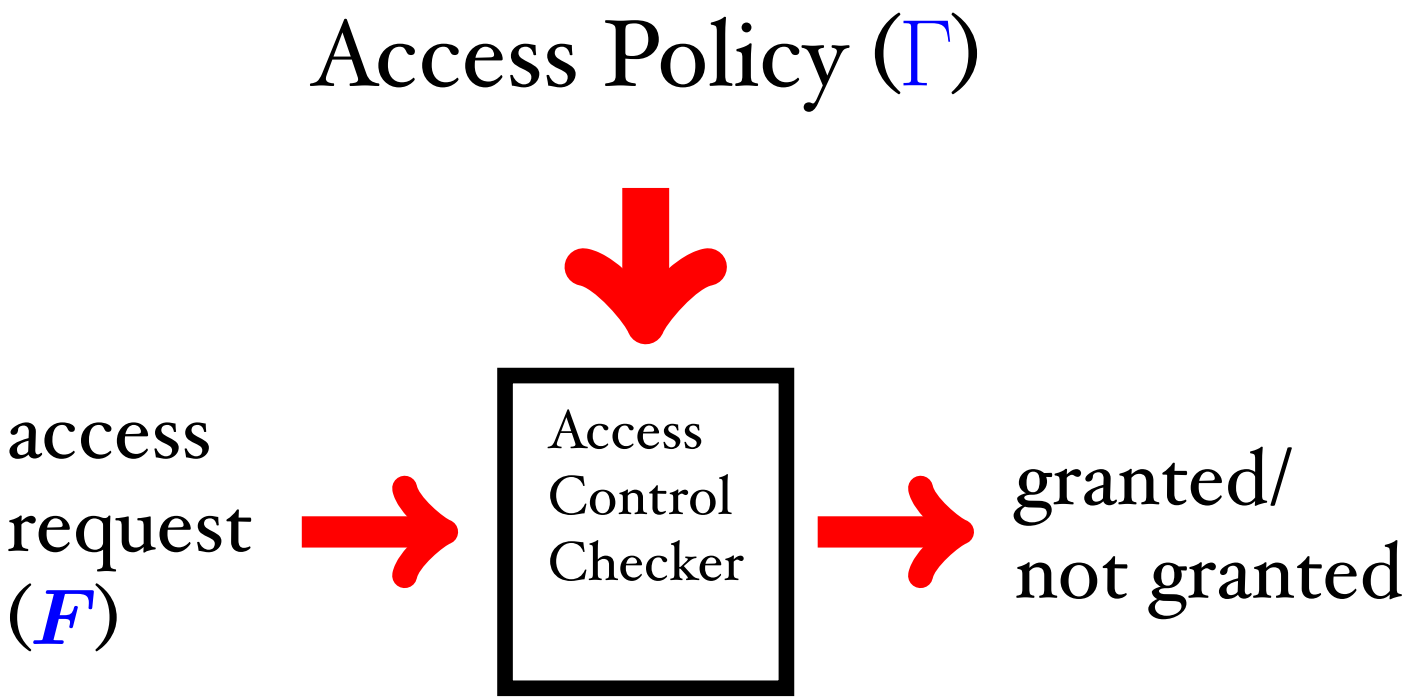
$$
\begin{aligned}
A \rightarrow B&: \quad N_a \\
B \rightarrow A&: \quad \{N_a, N_b\}_{K_{ab}} \\
A \rightarrow B&: \quad N_b
\end{aligned}
$$

5th Lecture: Access Control Logic

- formulas
- judgements

5th Lecture: Access Control Logic

- formulas
- judgements



5th Lecture: Access Control Logic

entails sign

$$\Gamma \vdash F$$

a single formula

Gamma
stands for a collection of formulas
("assumptions")

5th Lecture: Inference Rules

premisses

$$\frac{\Gamma \vdash \boldsymbol{F}_1 \qquad \Gamma \vdash \boldsymbol{F}_2}{\Gamma \vdash \boldsymbol{F}_1 \wedge \boldsymbol{F}_2}$$

conclusion

6th Lecture: Privacy