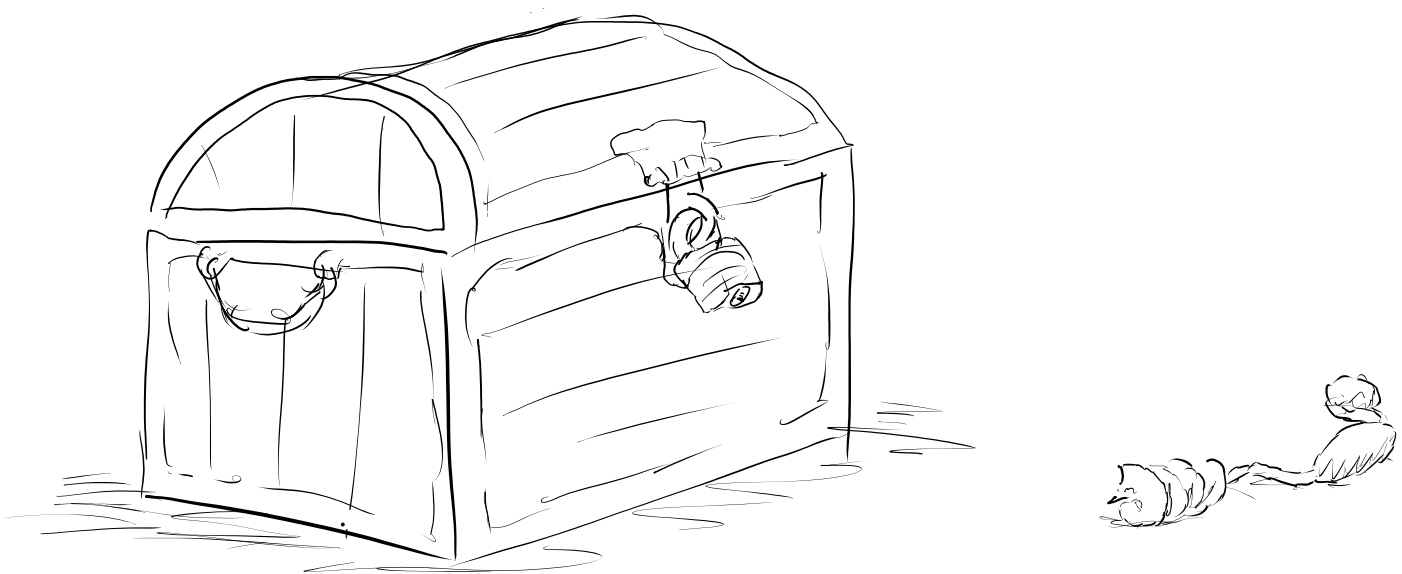


Upstream bug → vulnerability → exploit

allow an exploit

wrong person did wrong thing

bad sorting algorithm

1. find bug
2. find how used
3. plan attack

↳ vulnerability

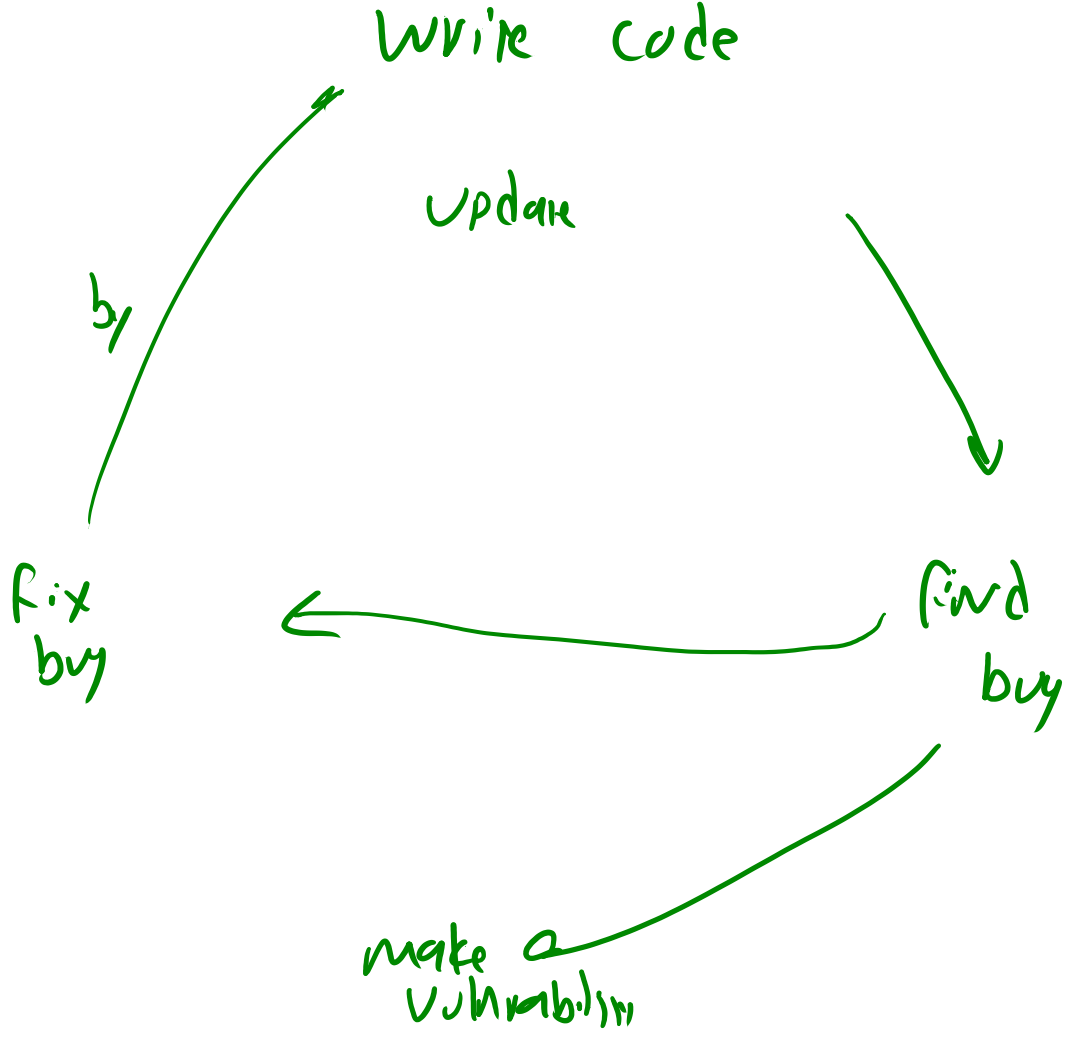
login:

Sort all known users

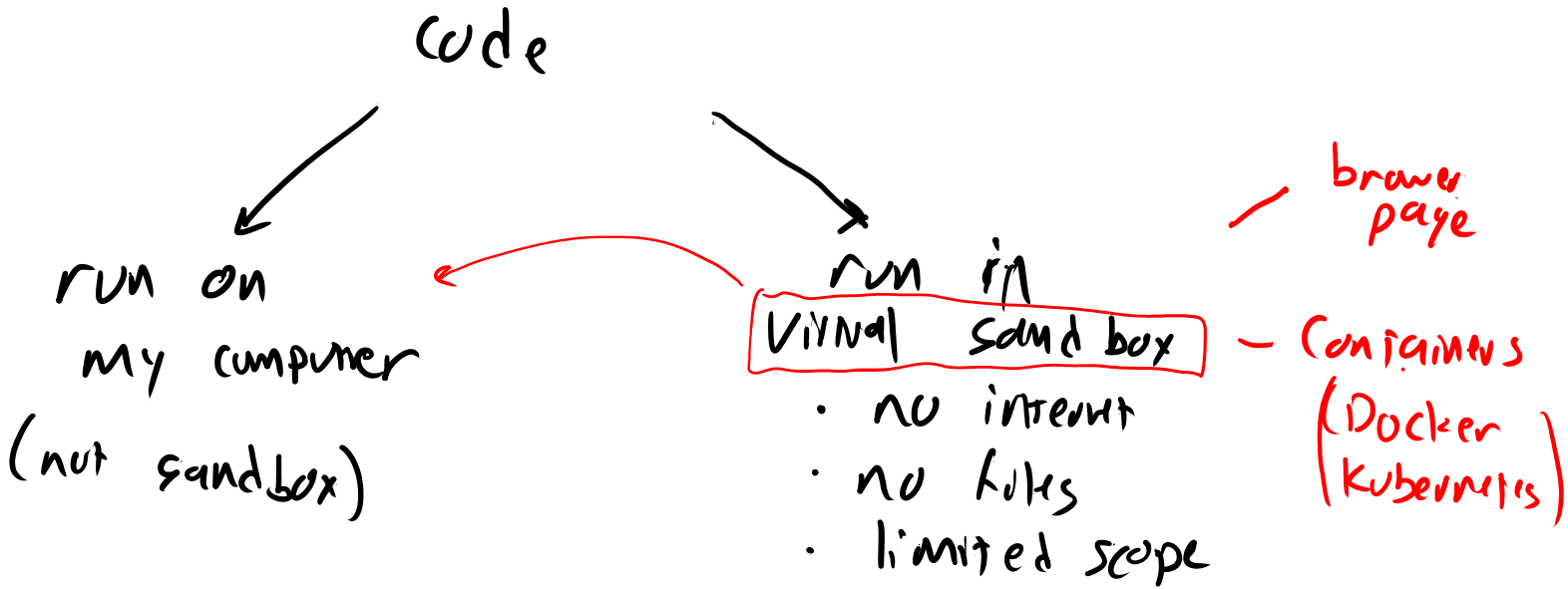
Check this user in list

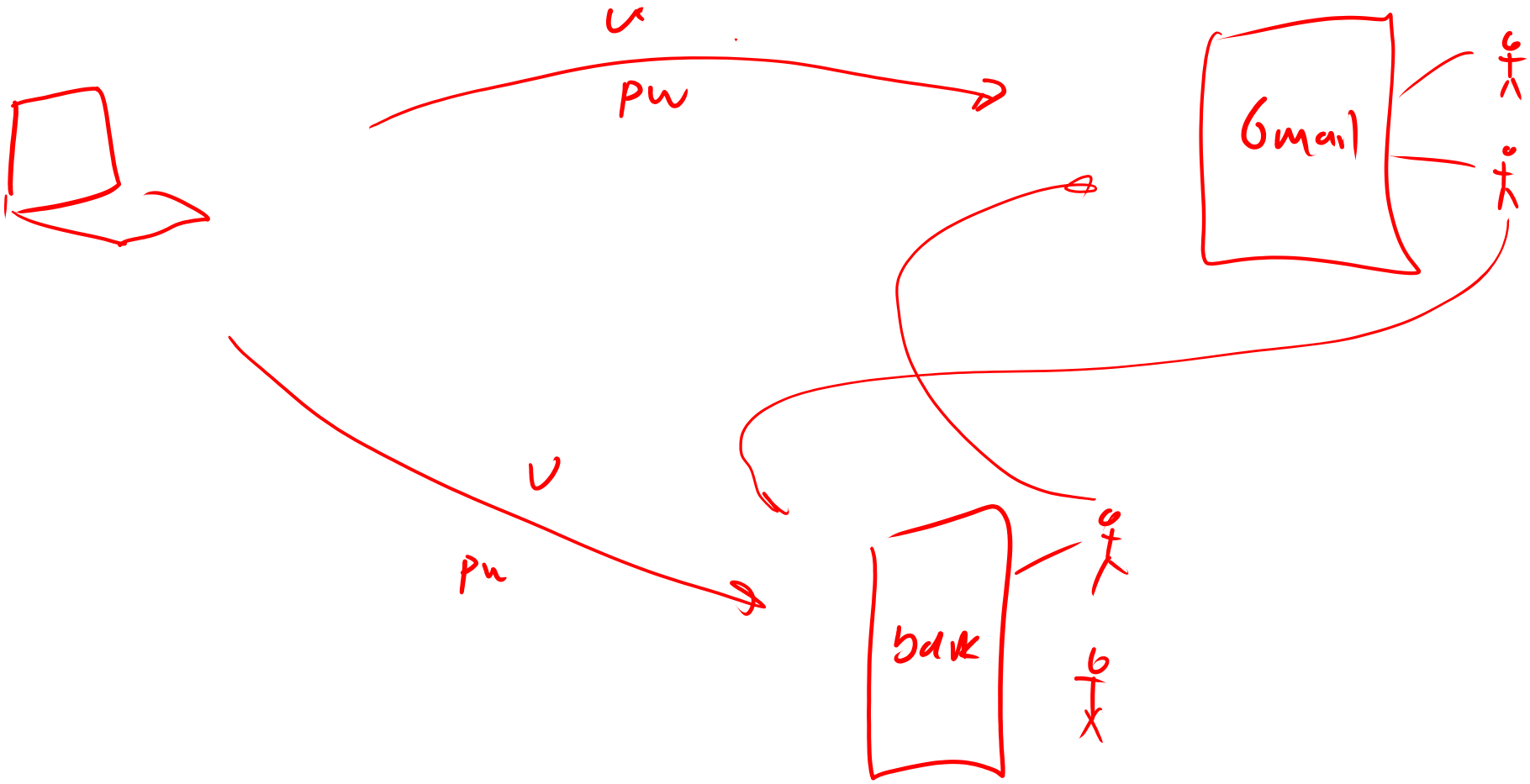
Attack:

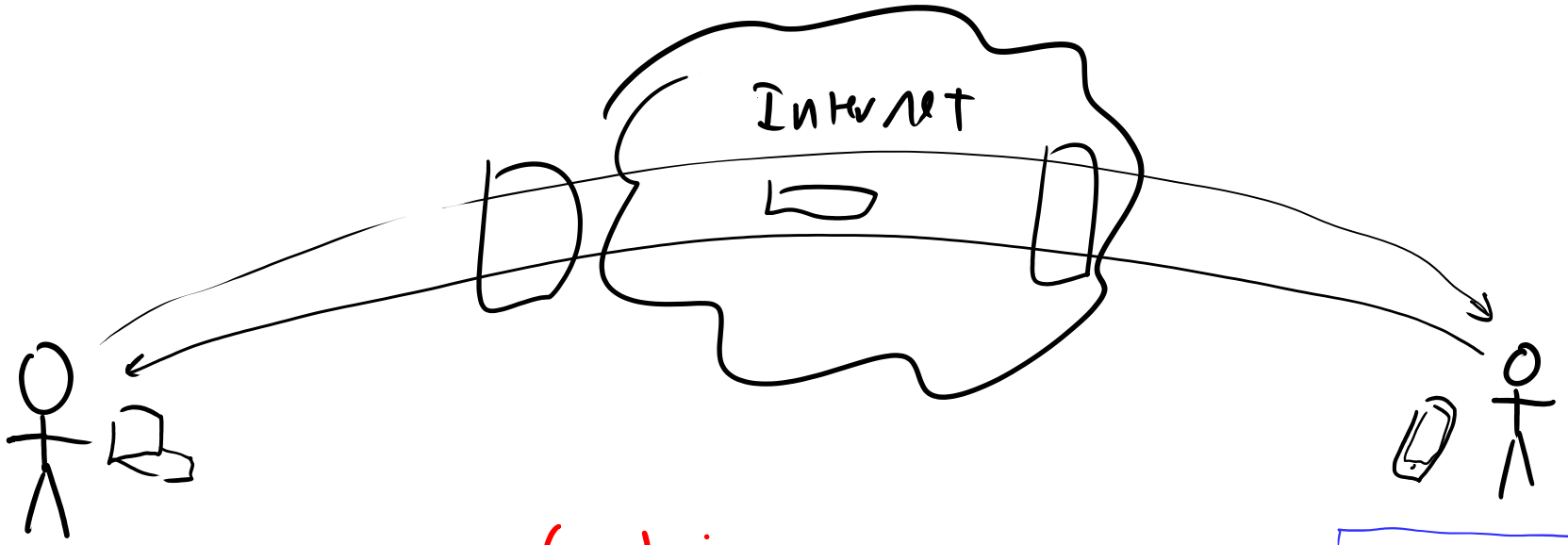
Creat many accounts in right order to miss-sort target account


write code
update
by
fix
bug
find
bug
make a
vulnerability

Sandboxing


code
run on
my computer
(not sandbox)
run in
virtual sand box
· no internet
· no files
· limited scope
browser page
– containers
(Docker
Kubernetes)


u
pw
Gmail
u
pw
bank

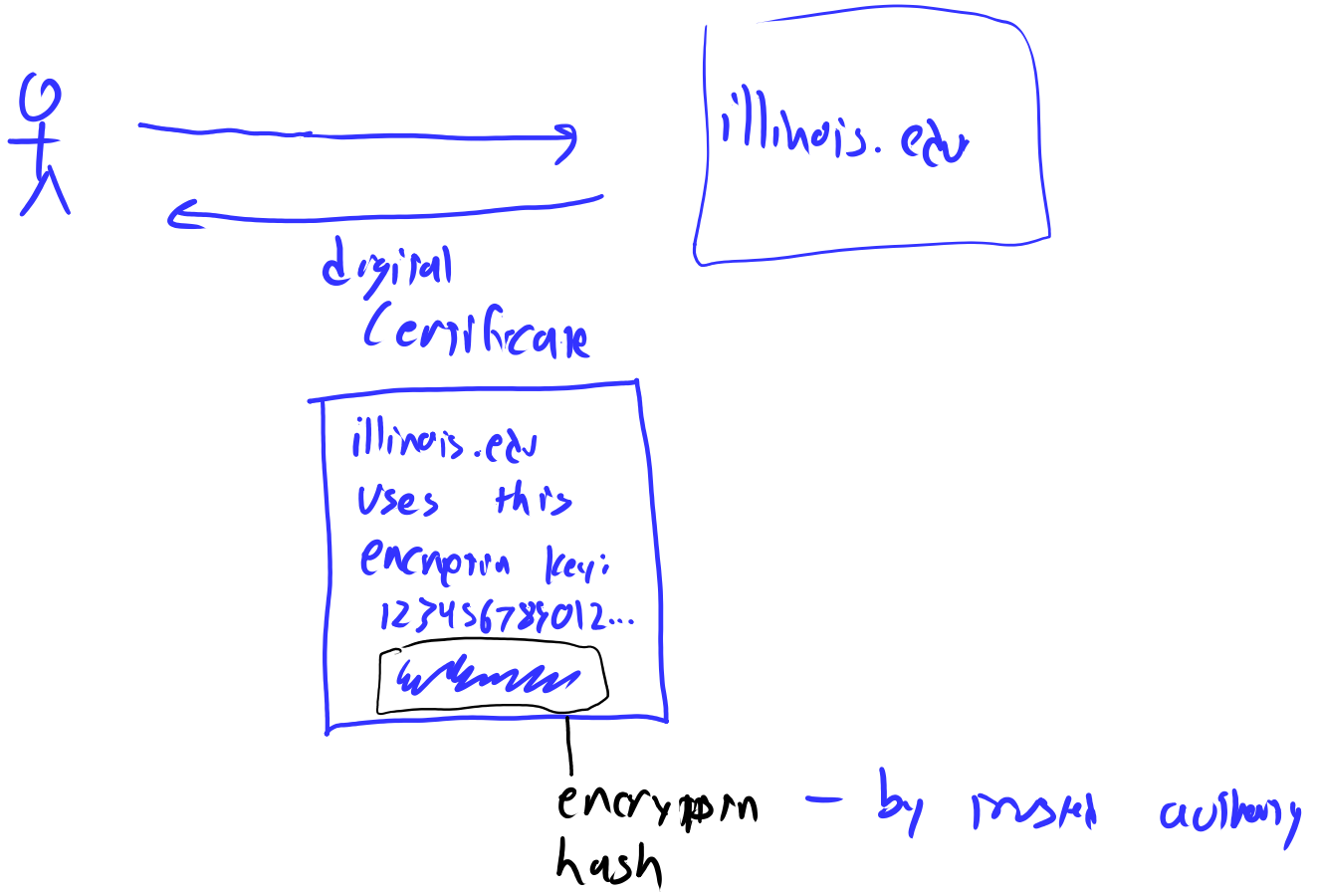


Goals:

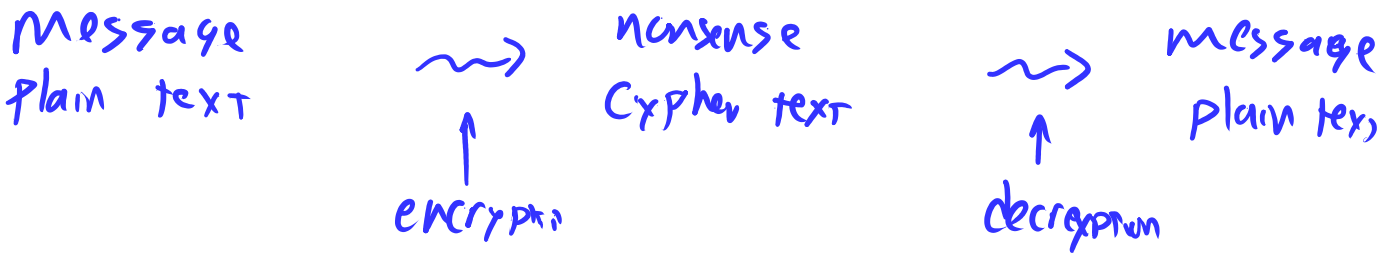
- talk to right person
- messages unmodified
- message unread by others
- no one knows we are talking

Authentication
Integrity
Confidentiality

Authentication



encryption



1. Public algorithm
2. private key

$f_1(m, k) \rightarrow c$

$f_2(c, k) \rightarrow m$

Hash

function in: any-length data

out: fixed-length data

property: find an input to get output is very hard

Sign: encrypt(hash(message), key)