CPSC 213 Lab 5

Memory Management & Reference Counting

Slides available at https://randyzhu.com/cpsc213

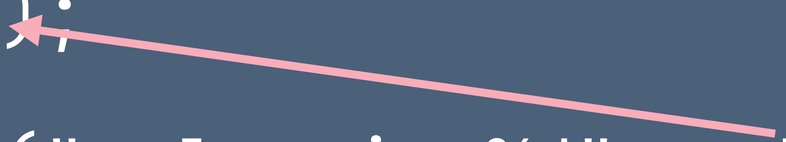
Randy Zhu

Course Updates

- Quiz 2 this week
  - Usual policy, no questions on quiz content
- Assignment 5 due this Friday, Feb 13th
  - Scary!
- Reading break next week, no course activities

Diagnose Memory Bugs (1.5)

```
void copy(int s, int* d):
  *d = s;
void foo(int s):
  int* d = malloc(sizeof(int));
  copy(s, d);
  free(d);
  printf("value is %d", *d);
```



- What's wrong?
  - Access to d after free
  - Dangling pointer
- free makes data invalid
- free'd memory should never be accessed

Diagnose Memory Bugs (1.2)

```
int* copy(int s):
  int* d = malloc(sizeof(int));
  *d = s;
  free(d);
  return d;

void foo(int s):
  int* d = copy(s);
  printf("value is %d", *d);
```

- What's wrong?
  - Free follows malloc
    - Looks fine?
  - We return free'd data

What's worse?

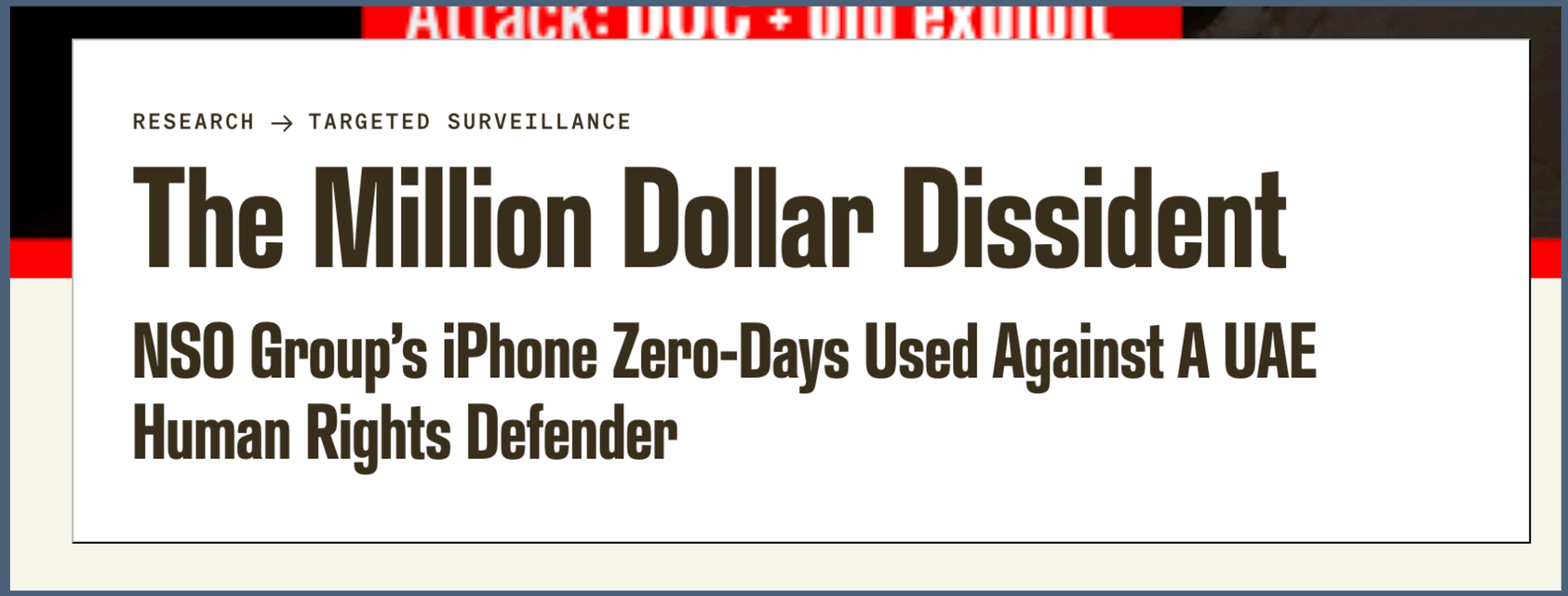
Dangling pointer vs memory leak

- Memory leaks
  - Worst case scenario, your code crashes
  - Violates resource management
    - Wasteful
  - Not end of the world if O(1) memory leaked
  - Defined behaviour

- Dangling pointers
  - Worst case scenario?

Trident: Safari UAF targets dissents

https://citizenlab.ca/research/million-dollar-dissident-iphone-zero-day-nso-group-uae/



Stagefright: UAF in Android MMS Stack

https://www.zdnet.com/article/stagefright-just-how-scary-is-it-for-android-users/

The really sneaky part is you don't need to watch the playful cats. If you're using Google's Hangouts app, you don't even need to open your text message app. All the attacker needs to do is send a poisoned package to your phone number. It then opens up your device, and the attack starts. This can happen so fast that by the time your phone alerts you that a message has arrived, you've already been hacked. If, on the other hand, you're using

What's worse?

Dangling pointer vs memory leak

- Memory leaks
  - Worst case scenario, your code crashes
  - Violates resource management
    - Wasteful
  - Not end of the world if O(1) memory leaked
  - Defined behaviour

- Dangling pointers
  - Worst case scenario?
- Introduces undefined behaviour
- What does your your program does after a accessing a dangling pointer?
  - **Nobody knows!**

refcount problem

Have a good reading break!

- Make sure to take care of yourselves