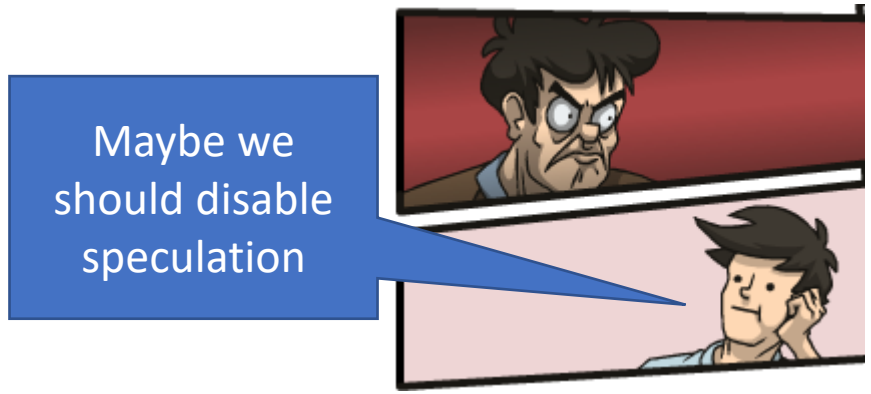
Dynamic Analysis for Spectre Mitigation

Alon Berkenstadt, Yakir Vizel, Mark Silberstein

Technion

Motivation

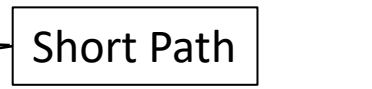
- Spectre is a speculative vulnerability
  - Causing data leakage
- Speculation is great for performance
  - Disabling it is not an option
- Data leakage is a bug
  - Security bugs are not welcomed
- How can we achieve safe speculation?



Proposal

- Transform indirect calls to something better
  - Allow benign speculative behavior
  - Ban unnecessary/buggy behaviors
  - **Profiling**

```
if (fun_ptr == foo)
        foo(x);
else if (fun_ptr == bar)
        bar(x);
else
        retpoline(fun_ptr,x);
```



Proposal

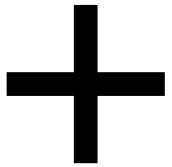
- Transform indirect calls to something better
  - Allow benign speculative behavior
  - Ban unnecessary/buggy behaviors

```
if (fun_ptr == foo)
        foo(x);
else if (fun_ptr == bar)
        bar(x);
else
        retpoline(fun_ptr,x);
```

Reduced the problem to Spectre V1

- SpecFuzz
- SpecTaint
- Etc.

Summary



- Mitigation for Spectre V1 and Spectre V2
  - Transform indirect calls
  - Adopt Spectre V1 mitigation
- Guidelines
  - Practical
  - Efficient
  - Security enhancing

Questions