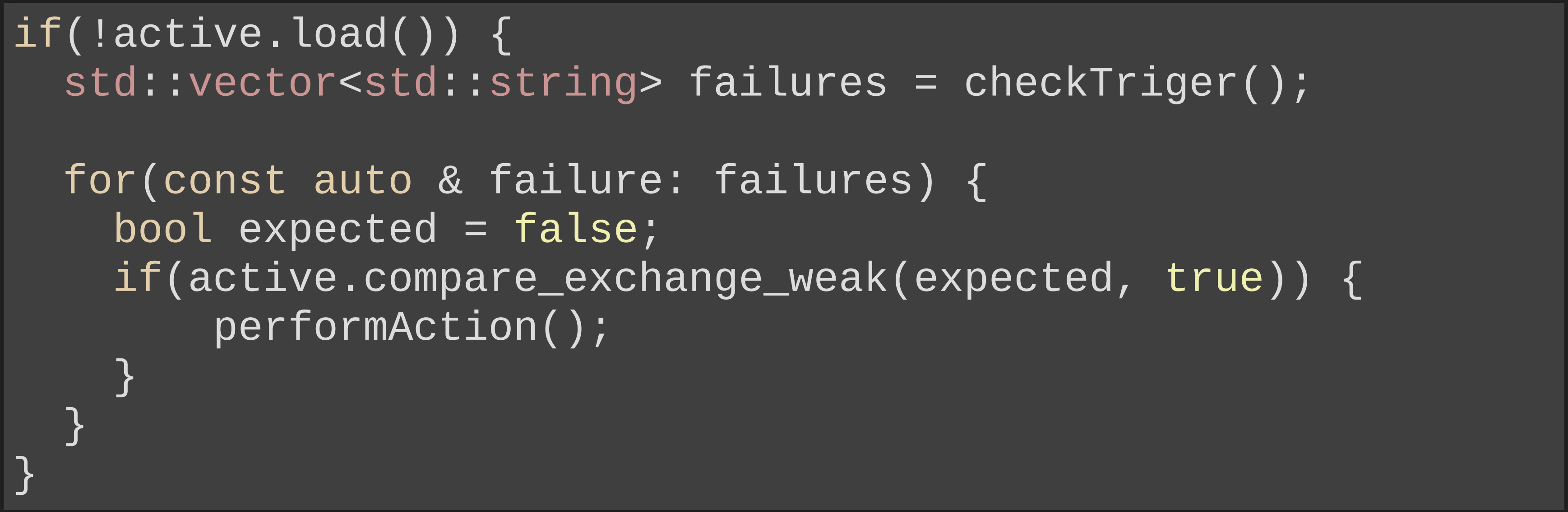
WHY RUST?

```
if(!active.load()) {
  bool expected = false;

  std::vector<std::string> failures = checkTriger();

  if(
    !failures.empty() &&
    active.compare_exchange_weak(expected, true)
  ) {
    performAction();
  }
}
```



```
if(!active.load()) {
  std::vector<std::string> failures = checkTriger();

  for(const auto & failure: failures) {
    bool expected = false;
    if(active.compare_exchange_weak(expected, true)) {
        performAction();
    }
  }
}
```

THATS WHY RUST

DISCLAIMER - REALLY RUST?

- For C(++) Devs: Yes
- For Java/C# Devs: Maybe
- For web backend Devs: Worth checking
- For web frontend Devs: Worh checking
- For functional Devs: Probably no
- For Go/D Devs: Come on, you found time for Go/D...

CARGO - GREAT TOOLSET

- Dependency tool
- Great code linter (cargo clippy)
- Generate your documentation (cargo doc)
- Keep your code formated (cargo fmt)
- UT integrated (cargo test)
- Keep your toolset updated (rustup)

> *70 percent of all security bugs are memory safety issues*

Catalin Cimpanu, Microsoft - Zero Day

Feb 11, 2019 - 15:48 GMT

STATIC ANALYSIS BUILD INTO LANGUAGE

AKA BORROW CHECKER

- No dangling pointers/iterators
- No `NullPointerException`
- No `SegmentationFault`
- No data races

AND ALL OF THIS WITH *NO GARBAGE COLLECTOR*

IT COMPILES TO NATIVE

IT MEANS IT IS REALLY PERFORMANT

SADLY ITS PROVEN, THAT SOMETIMES IT IS NOT THE BEST ON THIS FIELD

LEARNS FROM PREDECESSORS

- Great buildsystem
- Linkage level backward compatibility…
- … but syntax is a subject to evolve
- Inspired with FP languages, but still stays close to metal

TARGETS MANY FIELDS

- Native system language
- Easy FFI via C interfaces
- Works nice for embeded devices
- Easy to use REST/GraphQL crates
- Compiles to WASM, and easly integrates with JS
- Compiles to native GPU kernels (but still with issues)

ACTIVE COMMUNITY

- Rust Wrocław - visit our Slack!
- Rust users forum
- Rust internals forum
- Rust working groups
- This week in Rust
- Are we there yet?

GREAT RESOURCES

- The Book
- Rust by Example
- Rustling
- Cargo/Rustdoc Book
- Rustonomicon
- CmdLine/WASM/Embeded Book
- RustConf (and more) on YT

LEARN RUST

AND BECOME RUSTACEAN

VISIT US

RUST-WROCLAW.PL

GITHUB.COM/HASHEDONE

THANK YOU