YOLO

Frequently Resetting CPS for Security

Miguel A. Arroyo, M. Tarek Ibn Ziad, Hidenori Kobayashi, Junfeng Yang, Simha Sethumadhavan

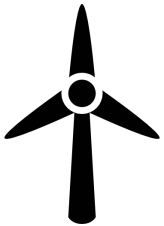
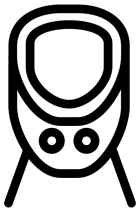
COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

YOLO

You **O**nly **L**ive **O**nce

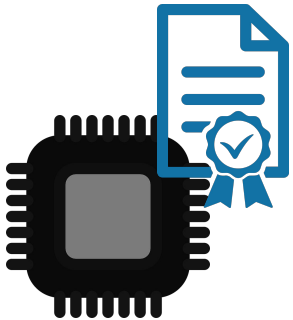


Cyber-Physical Systems = Cyber + Physical


CPS Characteristics (vs Cyber)

- More vulnerable to attacks
  - Not designed for security
  - Slow to no upgrades
- More difficult to recover from failures
  - Replacing hardware is non-trivial



CPS Characteristics (vs Cyber)

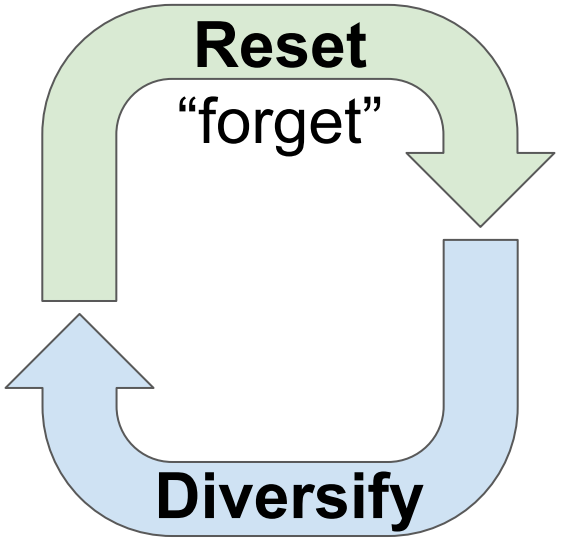
- Resilient by design
  - Redundancy against unintentional failures/faults

Key Research Question

Can we take advantage of unique CPS properties to protect them against security attacks?

YOLO in a nutshell

- Leverage ***physical*** characteristics of CPS to ensure ***cyber*** security.
- Flexible framework that can be integrated for a varying spectrum of systems.

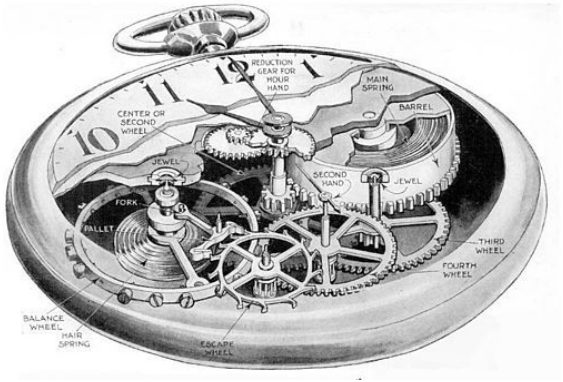


YOLO: Threat Model

- Attacker’s intention is to gain a foothold into the system.

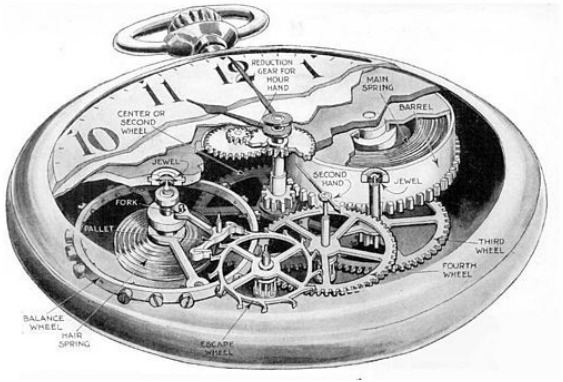
YOLO: Threat Model

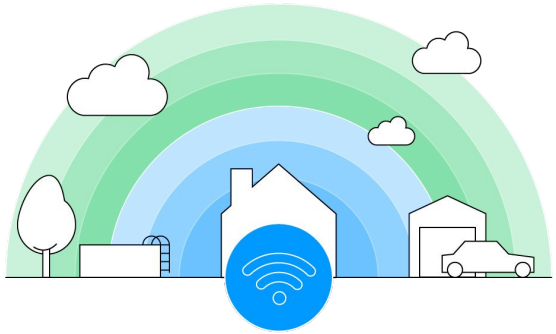
- Attacker’s intention is to gain a foothold into the system.
- An attacker has complete knowledge of the system internals.



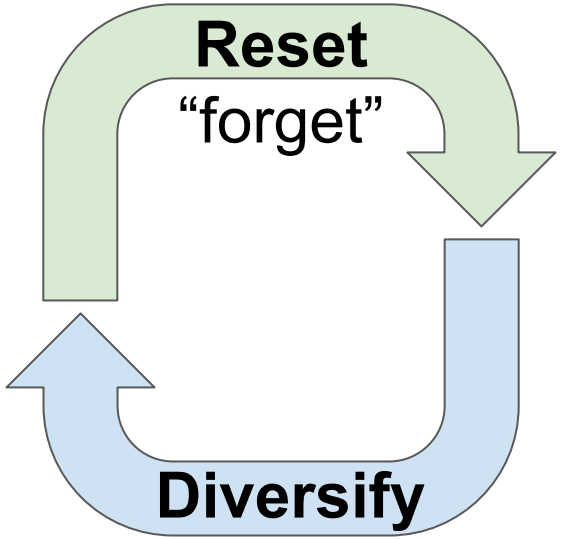
YOLO: Threat Model

- Attacker’s intention is to gain a foothold into the system.
- An attacker has complete knowledge of the system internals.
- An attacker’s sphere of influence is bounded.

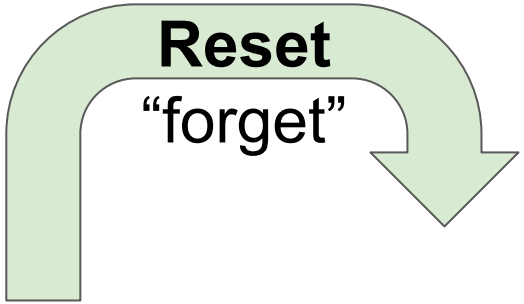




YOLO in a nutshell

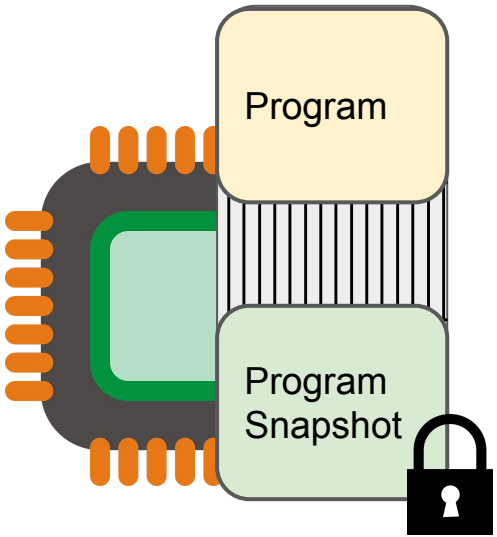


YOLO in a nutshell



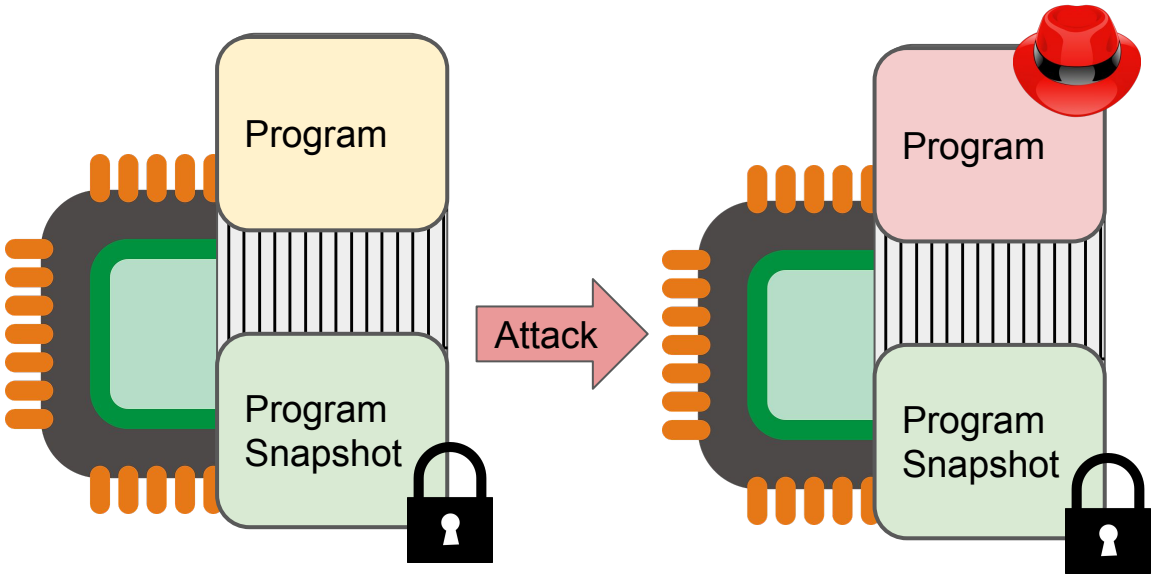
YOLO: You Only Live Once

- Why Reset?
  - Prevents an adversary's ability to corrupt the system.
    - Bounded time horizon over which an attacker can affect the system.



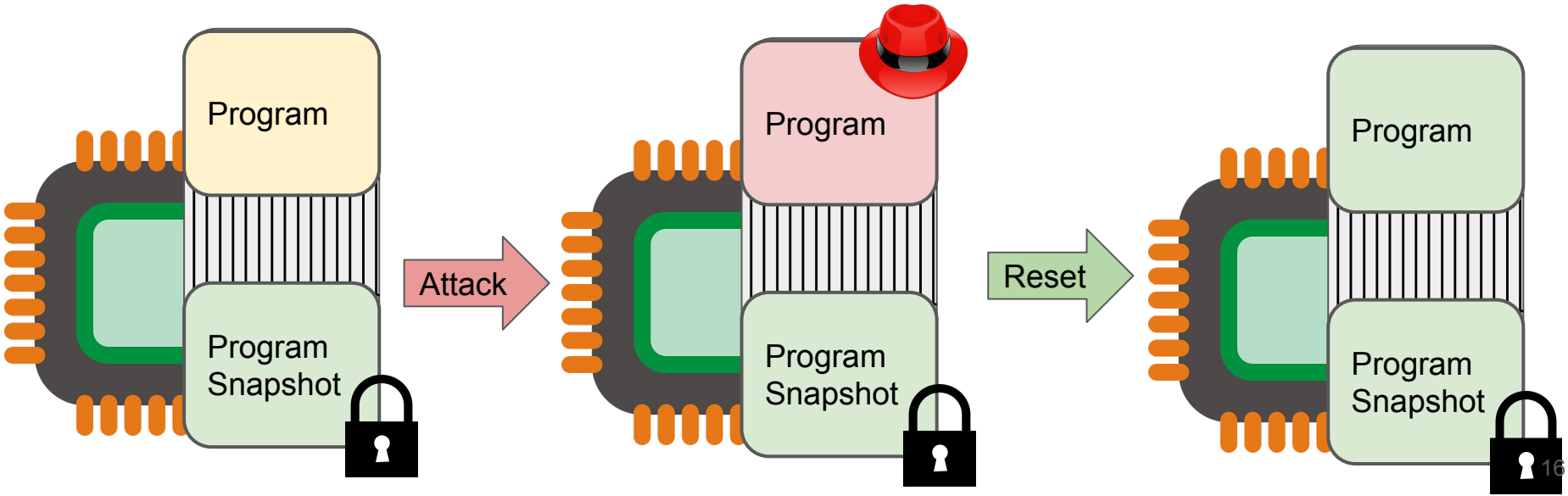
YOLO: You Only Live Once

- Why Reset?
  - Prevents an adversary's ability to corrupt the system.
    - Bounded time horizon over which an attacker can affect the system.

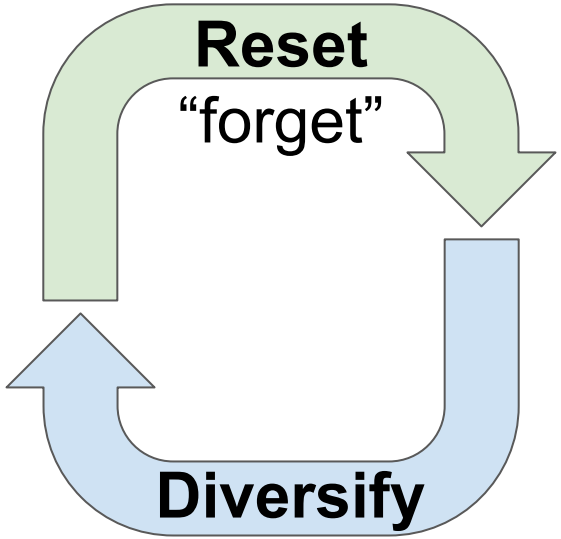


YOLO: You Only Live Once

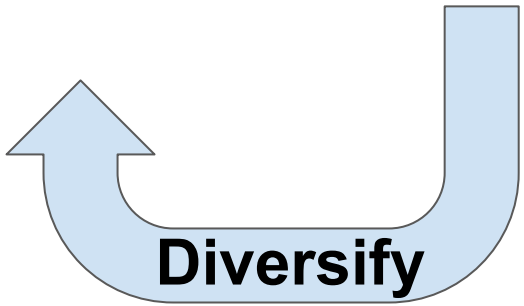
- Why Reset?
  - Prevents an adversary's ability to corrupt the system.
    - Bounded time horizon over which an attacker can affect the system.



YOLO in a nutshell

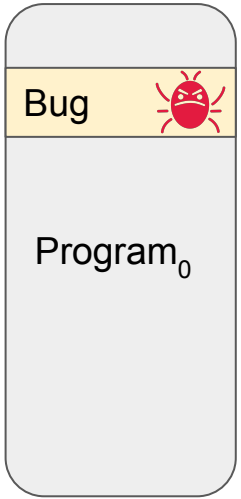


YOLO in a nutshell



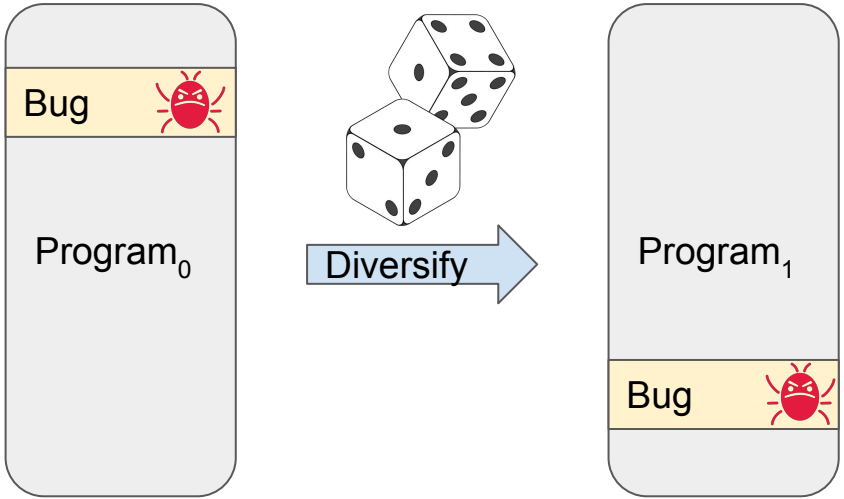
YOLO: You Only Live Once

- Why Diversify?
  - Introduce randomness to prevent the system from being compromised by the same method continuously.
    - Reduce chance of attacker success.



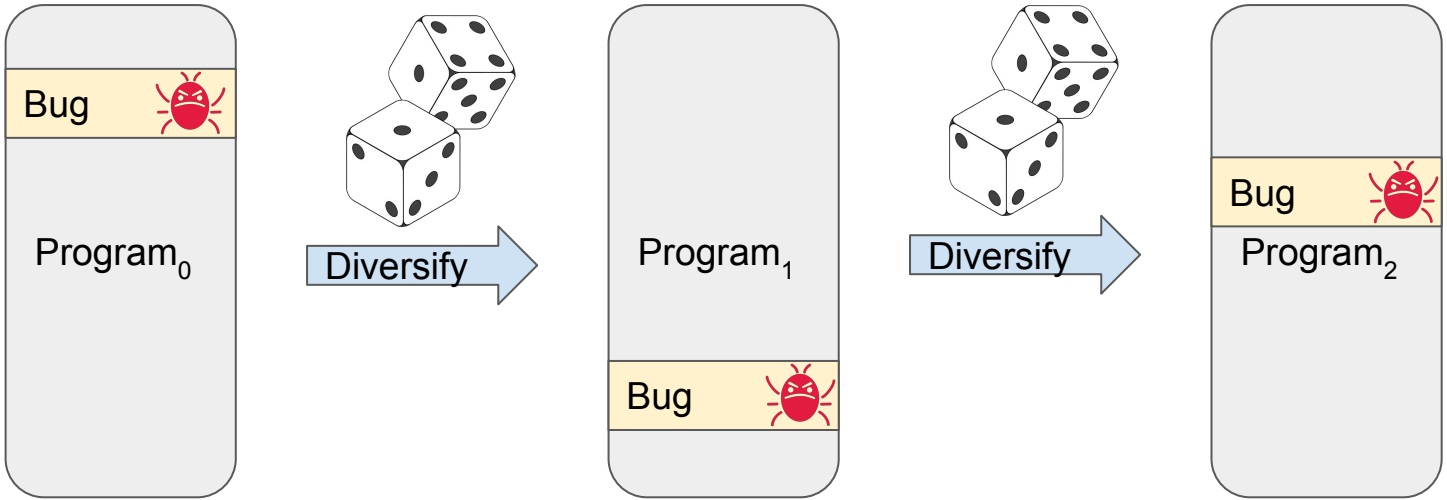
YOLO: You Only Live Once

- Why Diversify?
  - Introduce randomness to prevent the system from being compromised by the same method continuously.
    - Reduce chance of attacker success.



YOLO: You Only Live Once

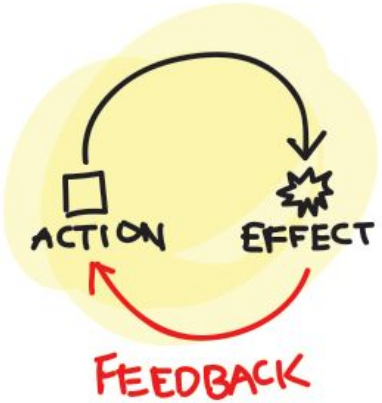
- Why Diversify?
  - Introduce randomness to prevent the system from being compromised by the same method continuously.
    - Reduce chance of attacker success.



YOLO: You Only Live Once

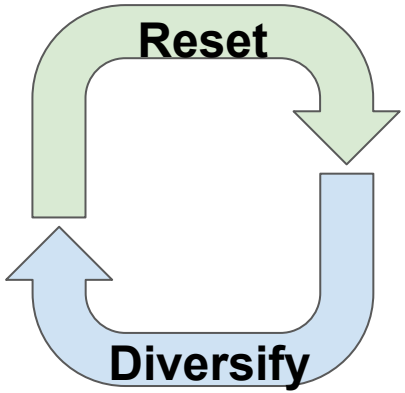
- Why does this work for CPS?

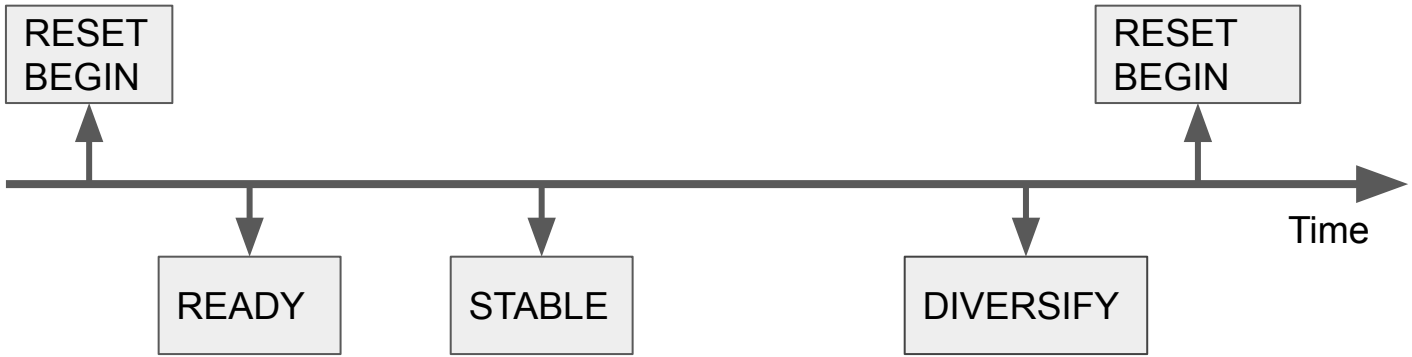
Inertia
Allows system to continue operation.



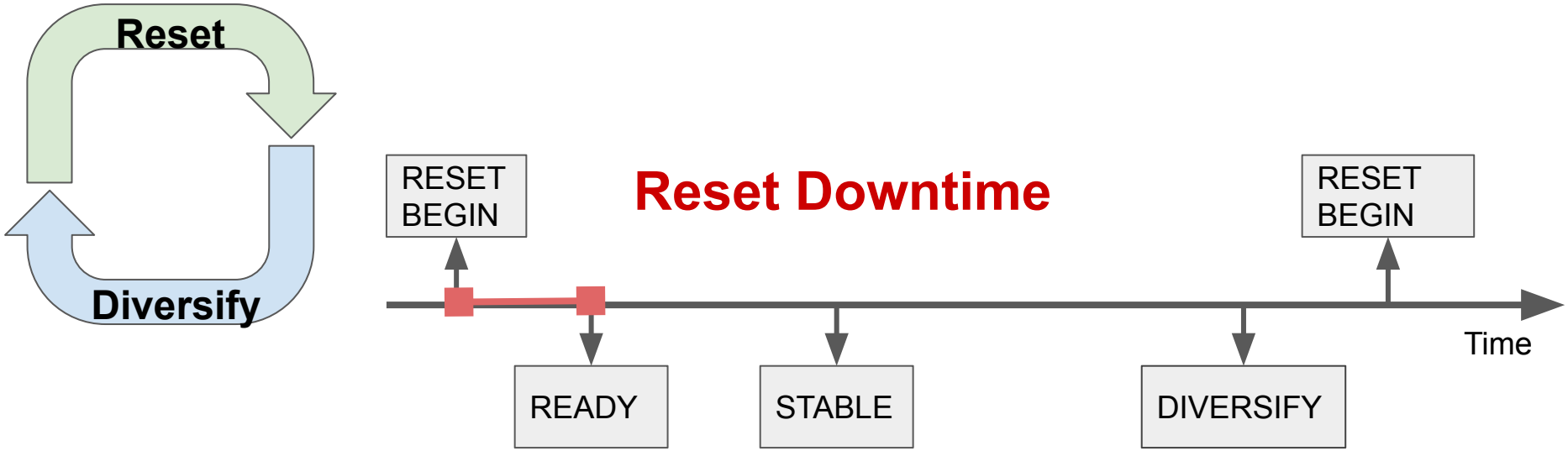
Feedback
The state of the system can be observed.

Why does **YOLO** provide protection?

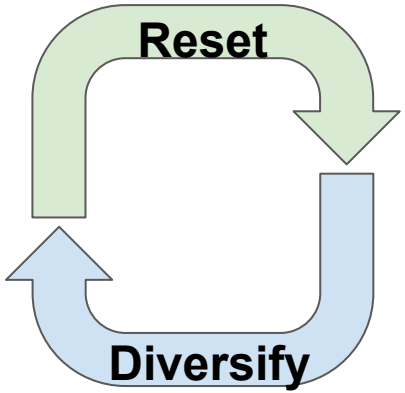


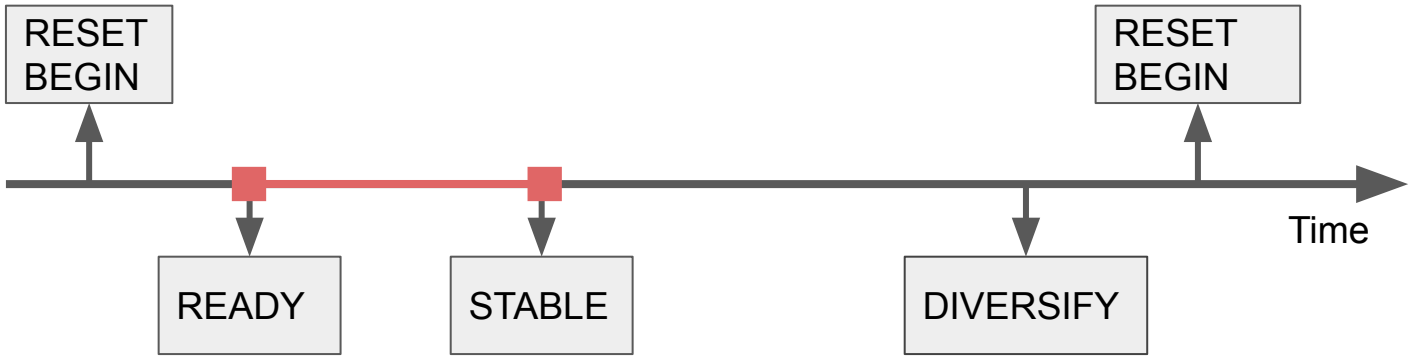


Why does **YOLO** provide protection?

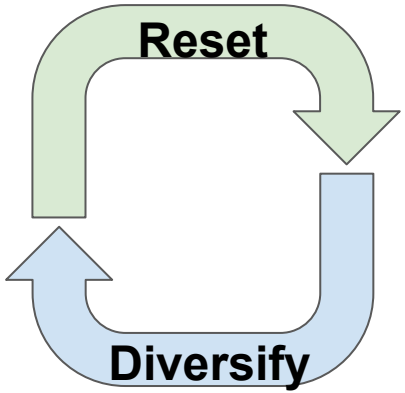


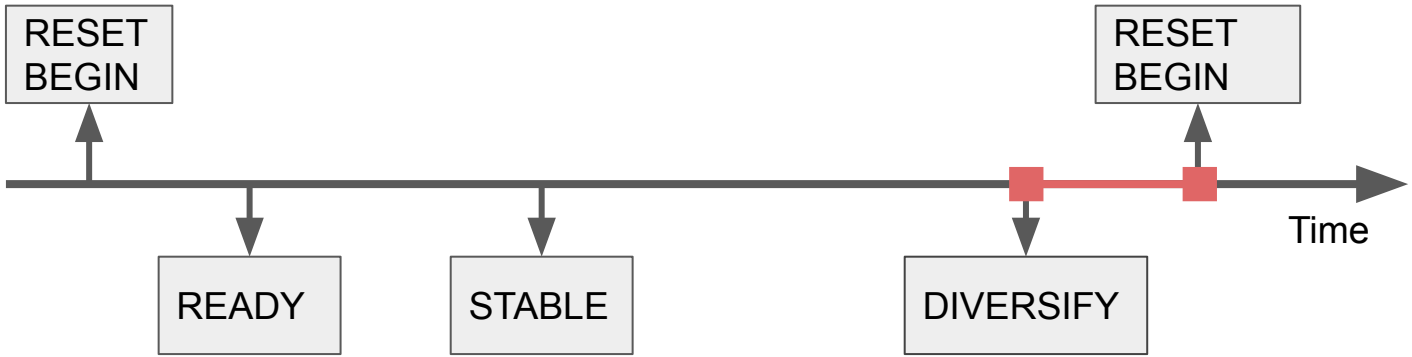
Why does **YOLO** provide protection?



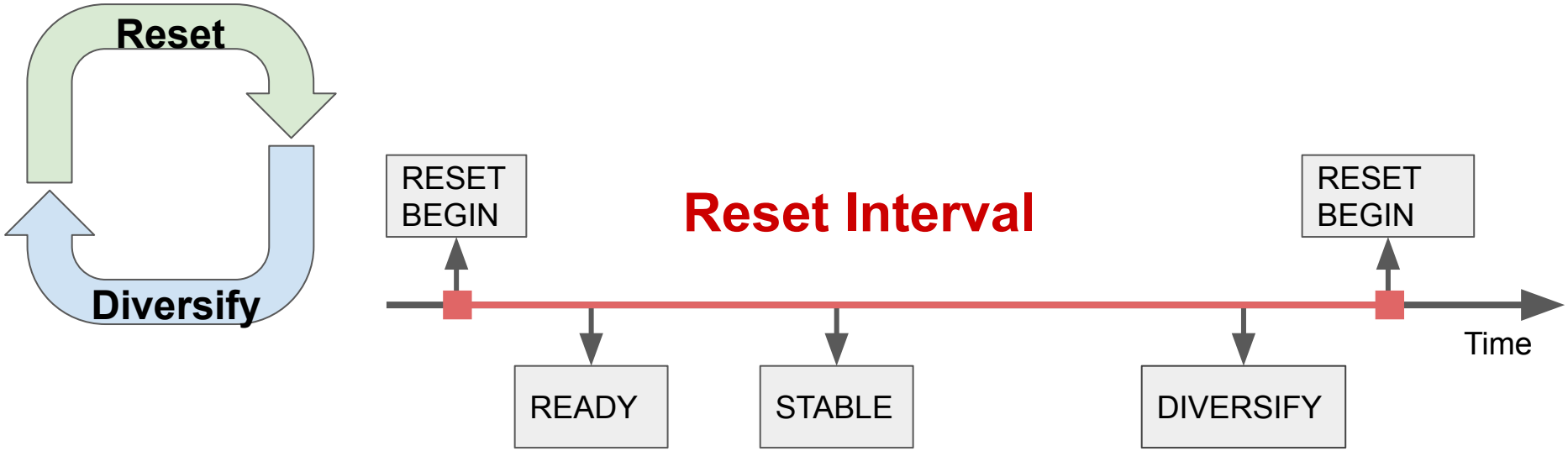


Why does **YOLO** provide protection?



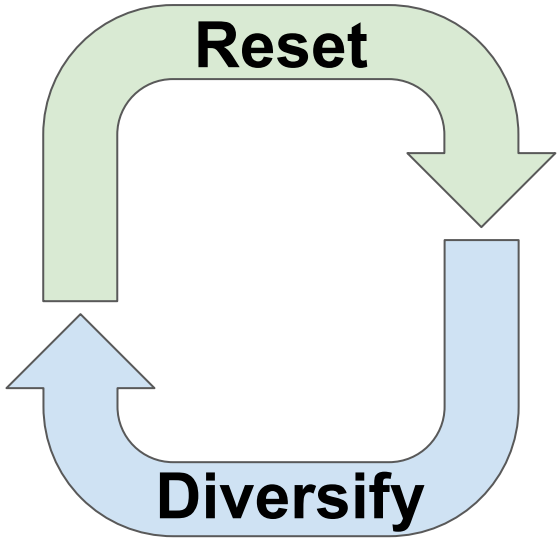


Why does **YOLO** provide protection?



- For YOLO to win: reset interval **<** time for an attacker's effects to manifest.

Why does **YOLO** provide protection?



- Persistent malware is denied (**RESET** step)
  - Memory is wiped clean.
- Increased work for the attacker (**DIVERSIFY** step)
  - Inputs have to be crafted to exploit each variant.

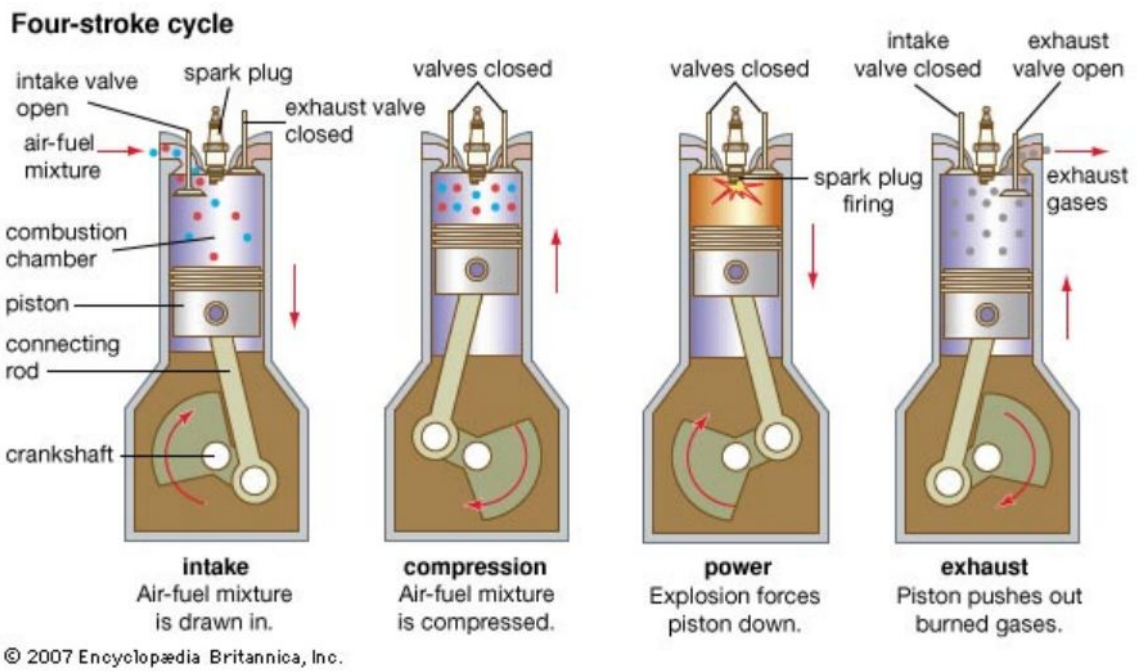
Rest of the talk...

Case Study 1: Engine Control Unit (ECU)

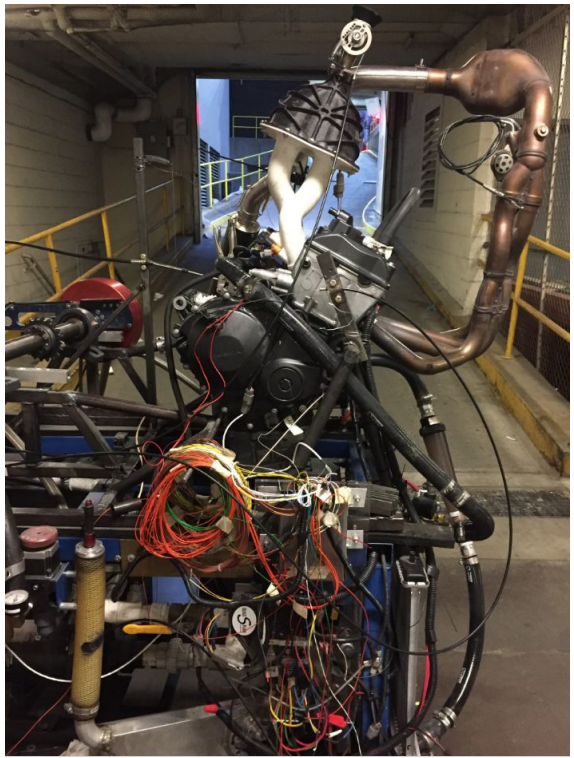
Case Study 2: Flight Controller (FCU)

Case Study - **ECU**

How it works



Case Study - **ECU**



- rusEFI: Open Source ECU
  - C/C++
- Honda CBR600RR Engine
- Cortex M4 @168 MHz
  - 192 KB SRAM
  - 1 MB Flash

Case Study - **ECU**

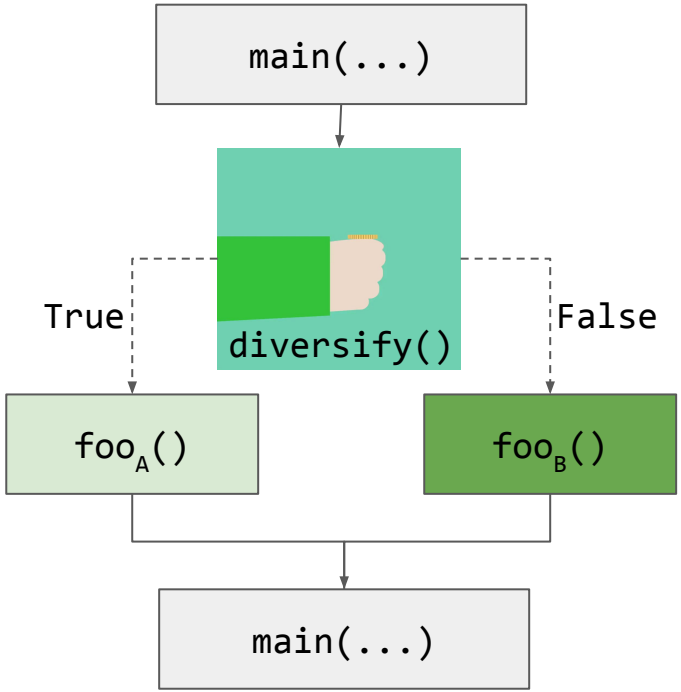
Reset Strategy

- Power cycle.
  - Externally triggerable.
  - Clears RAM & peripheral state.

Case Study - **ECU**

Diversify Strategy

- Build off technique called *Isomeron* [1].
  - Execution-path randomization.
  - Compile-time implementation.

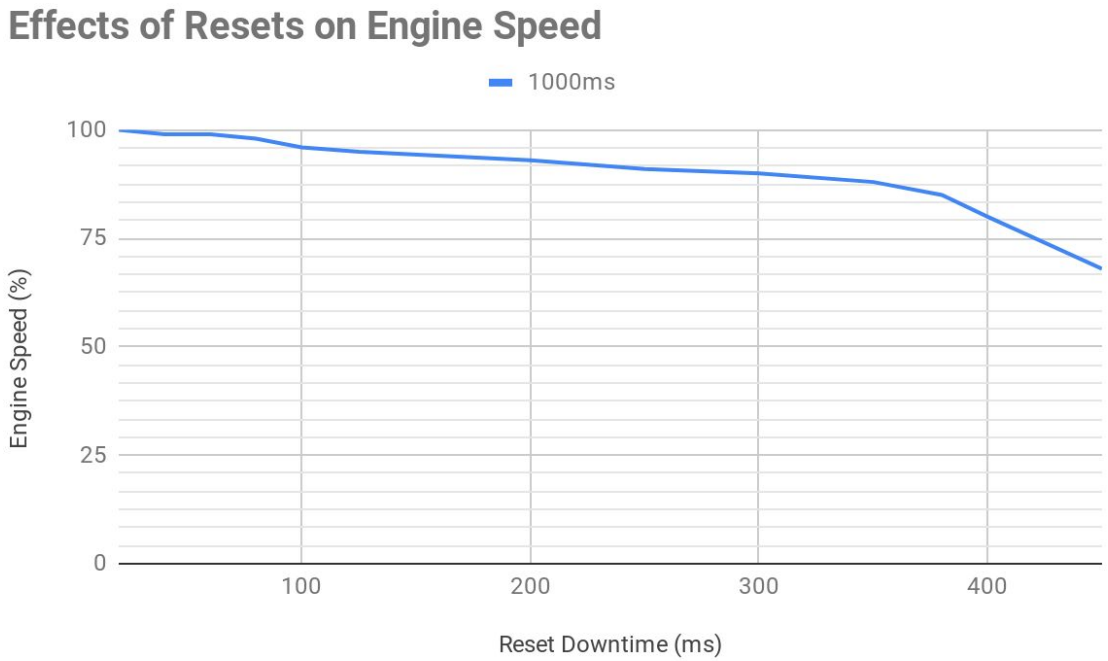


Program Control Flow Graph

[1] Davi, Lucas, et al. "Isomeron: Code Randomization Resilient to (Just-In-Time) Return-Oriented Programming." *NDSS*. 2015.

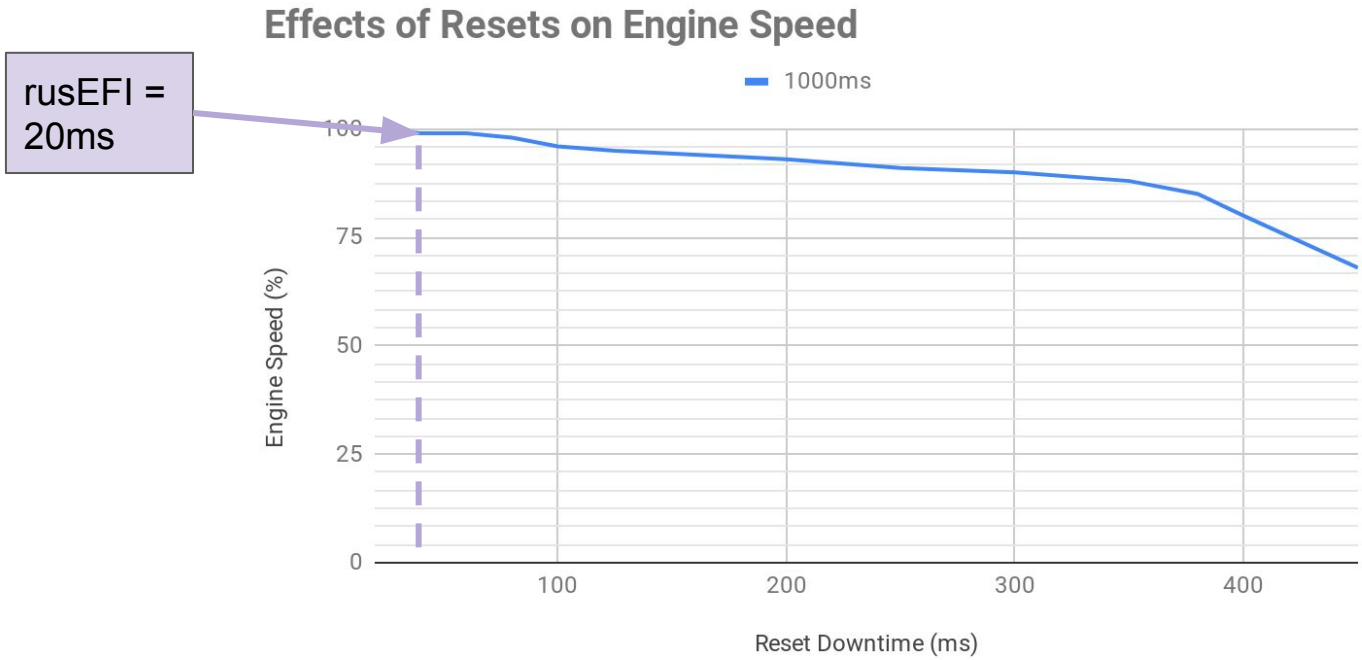
Case Study - **ECU**

YOLO Performance



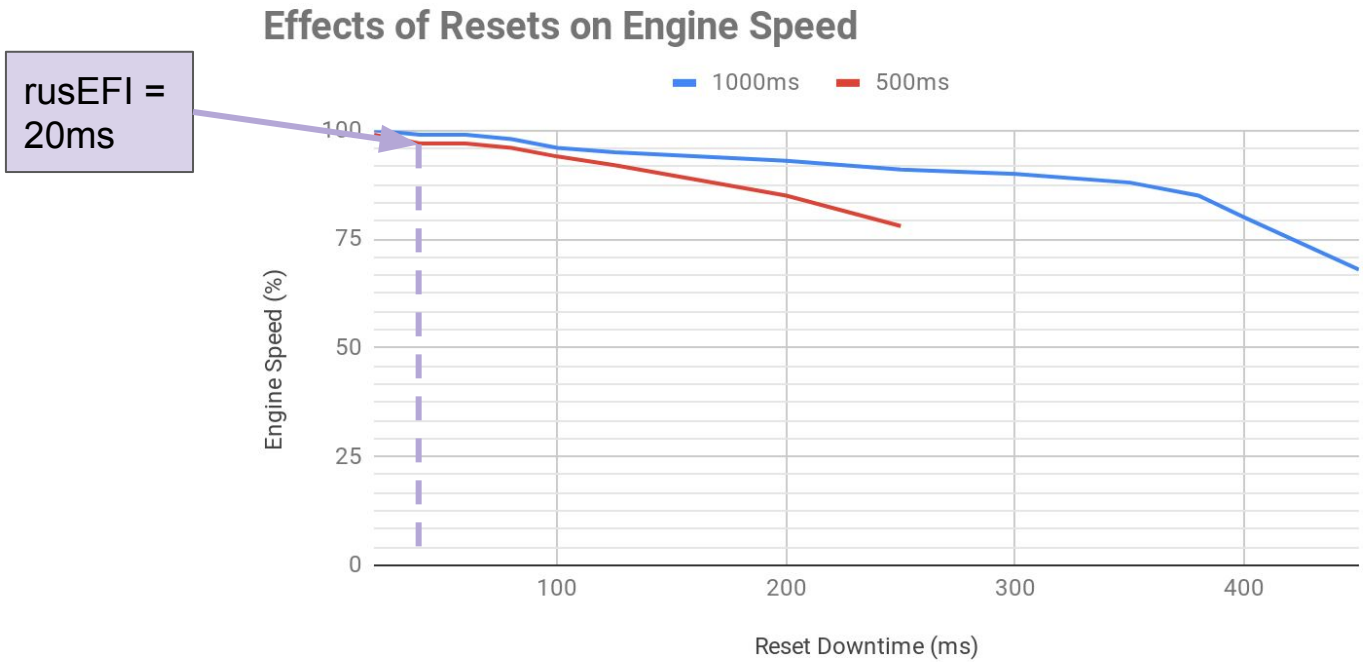
Case Study - **ECU**

YOLO Performance



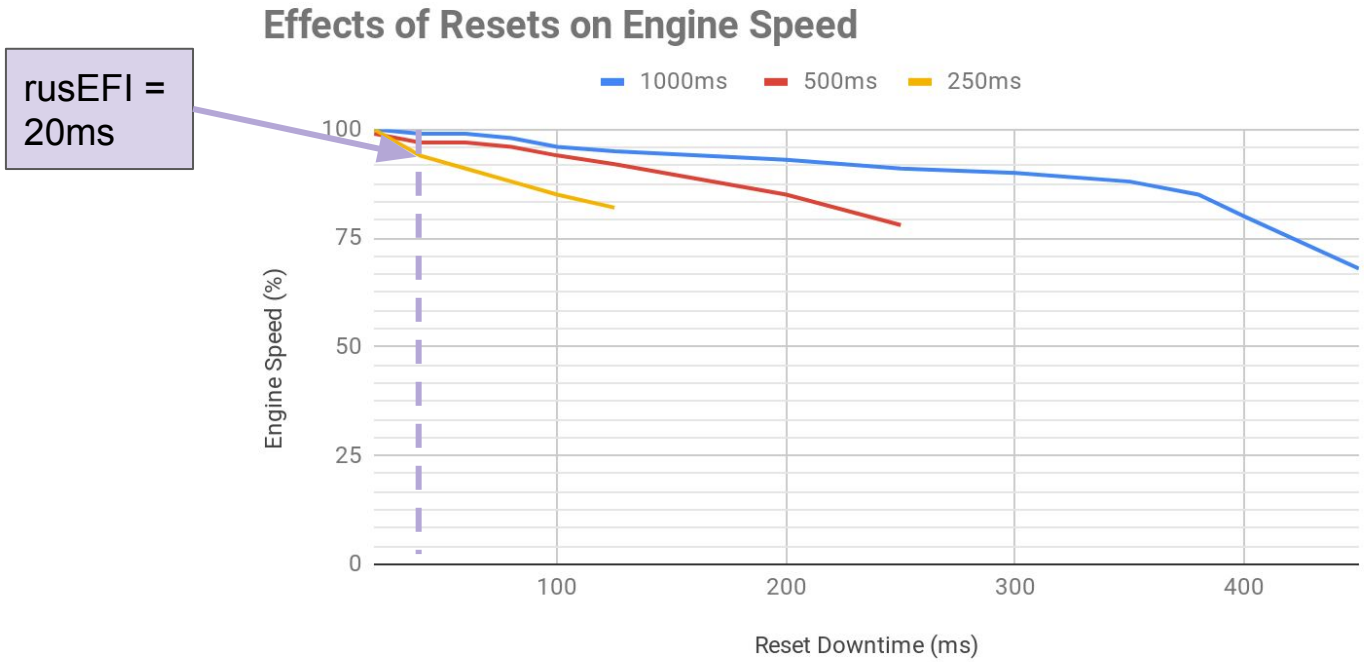
Case Study - **ECU**

YOLO Performance



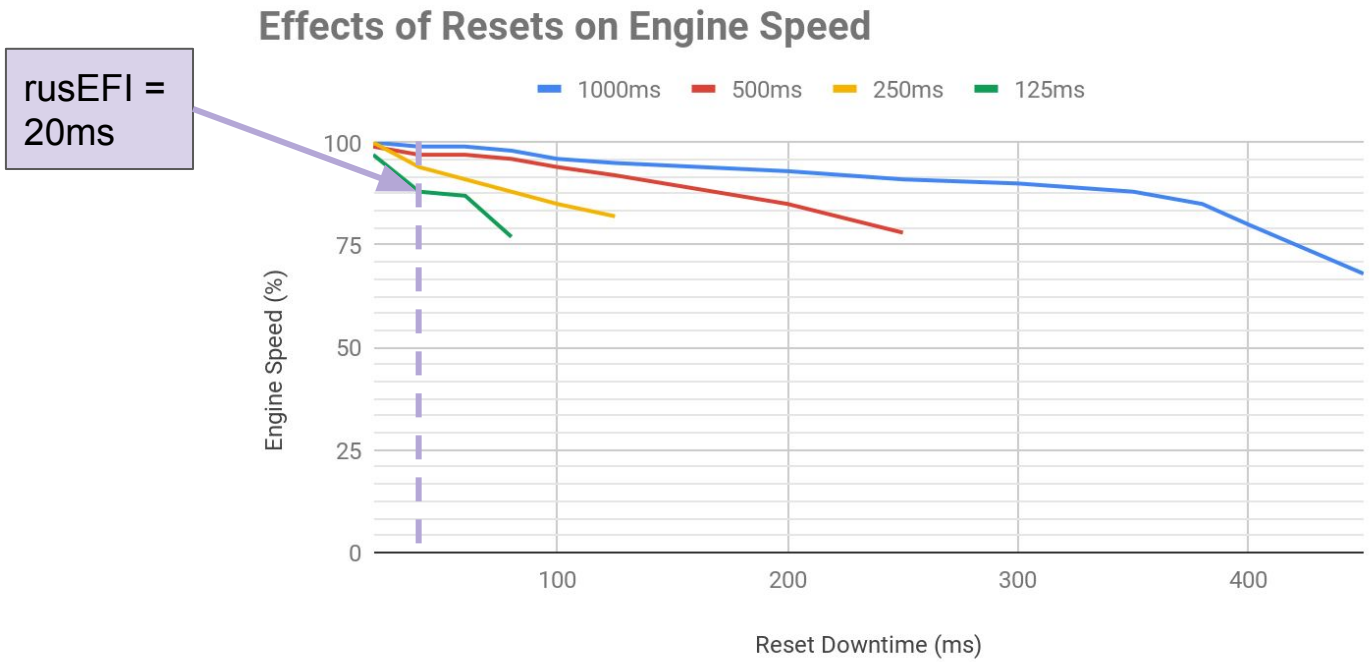
Case Study - **ECU**

YOLO Performance



Case Study - **ECU**

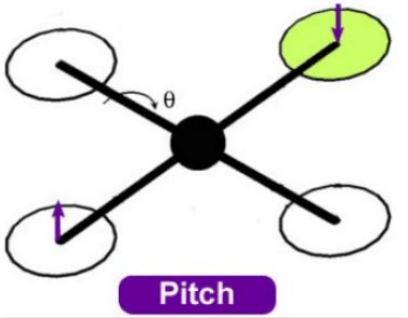
YOLO Performance

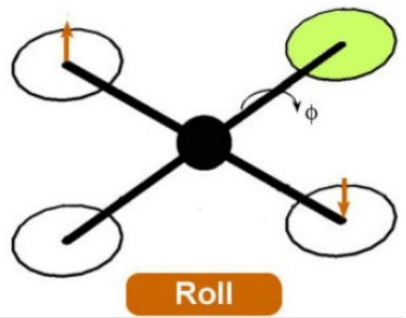


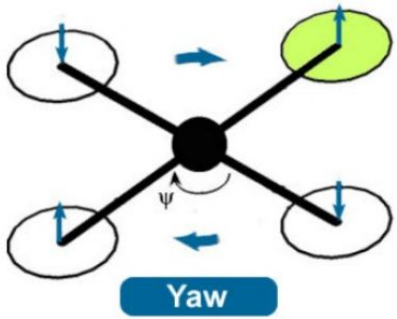
Case Study - **Flight Controller**

Case Study - **Flight Controller**

How it works







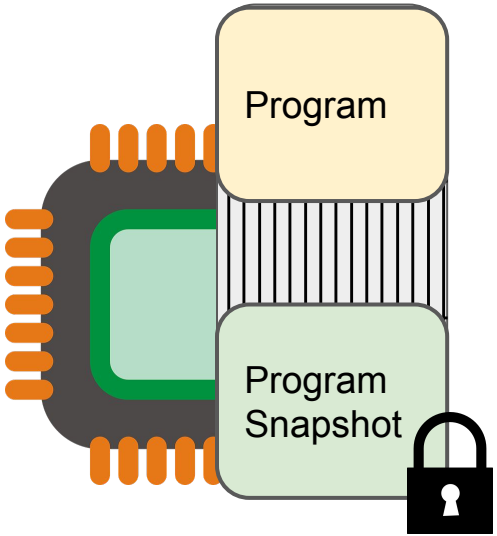
Case Study - **Flight Controller**

- PX4: Open Source FC
  - C/C++
- DJI F450 Flamewheel
- Cortex M4 @168 MHz
  - 192 KB SRAM
  - 1 MB Flash

Case Study - **Flight Controller**

Reset Strategy

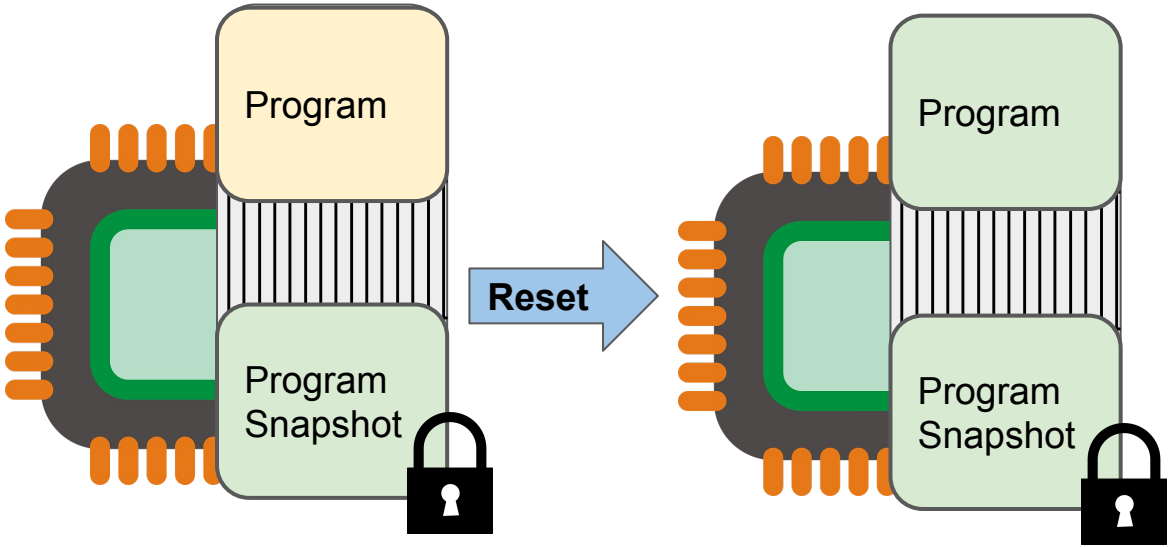
- Snapshot & Restore
  - Pre-initialized state for fast startup



Case Study - **Flight Controller**

Reset Strategy

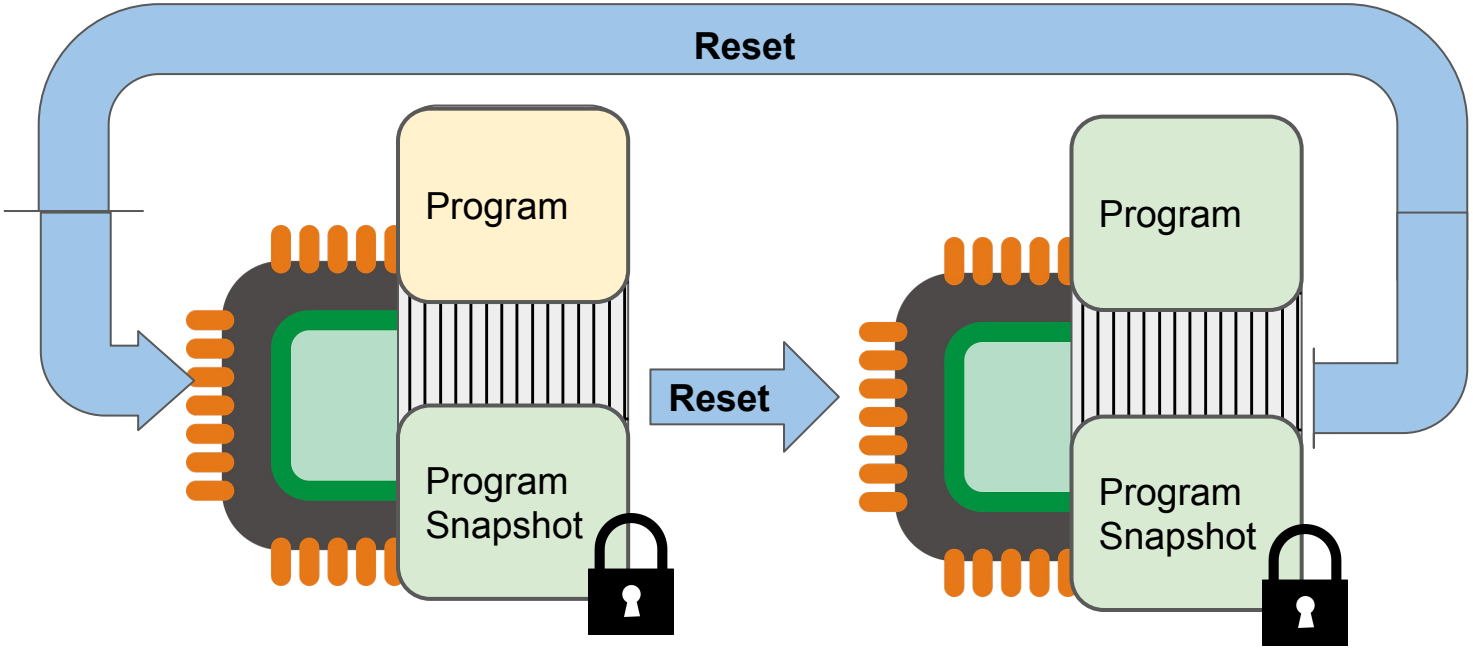
- Snapshot & Restore
  - Pre-initialized state for fast startup



Case Study - **Flight Controller**

Reset Strategy

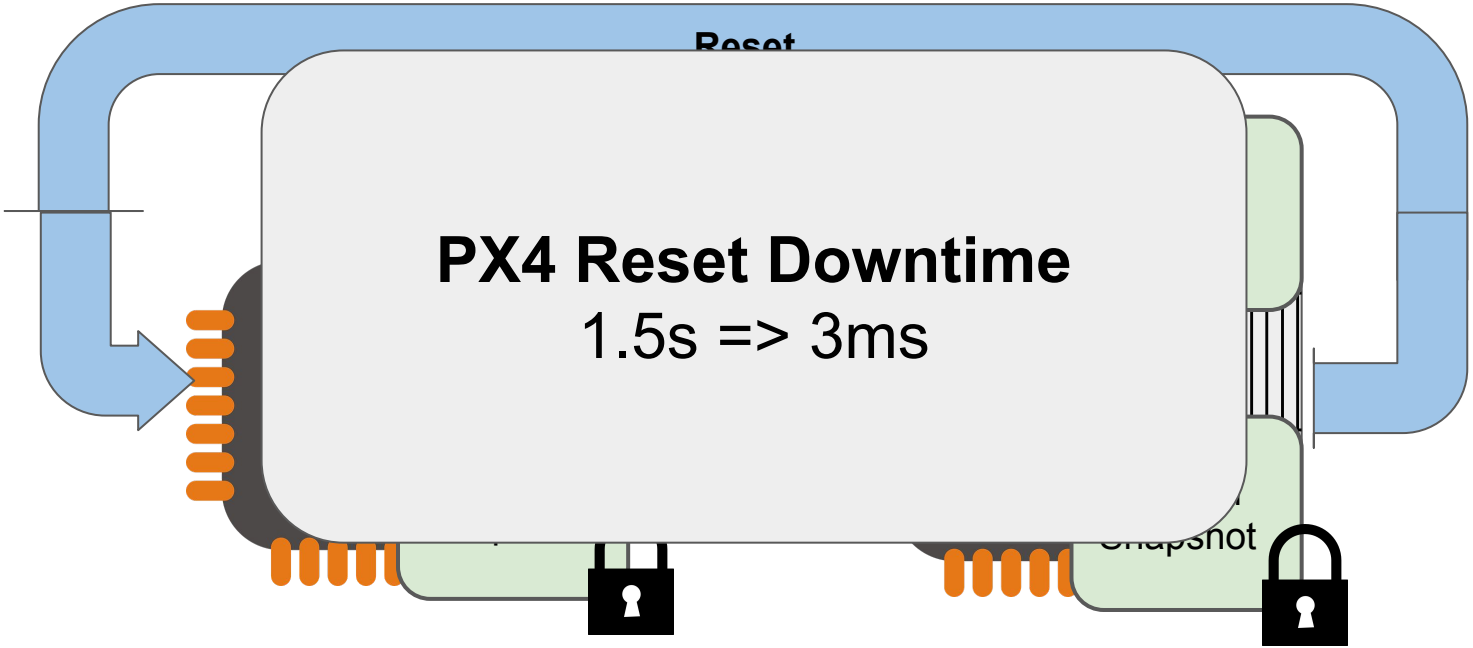
- Snapshot & Restore



Case Study - **Flight Controller**

Reset Strategy

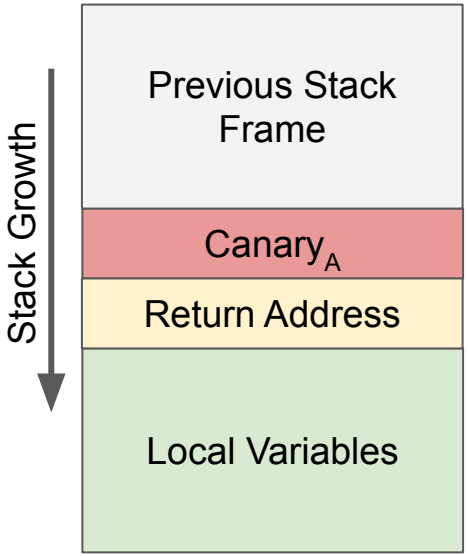
- Snapshot & Restore



Case Study - **Flight Controller**

Diversify Strategy

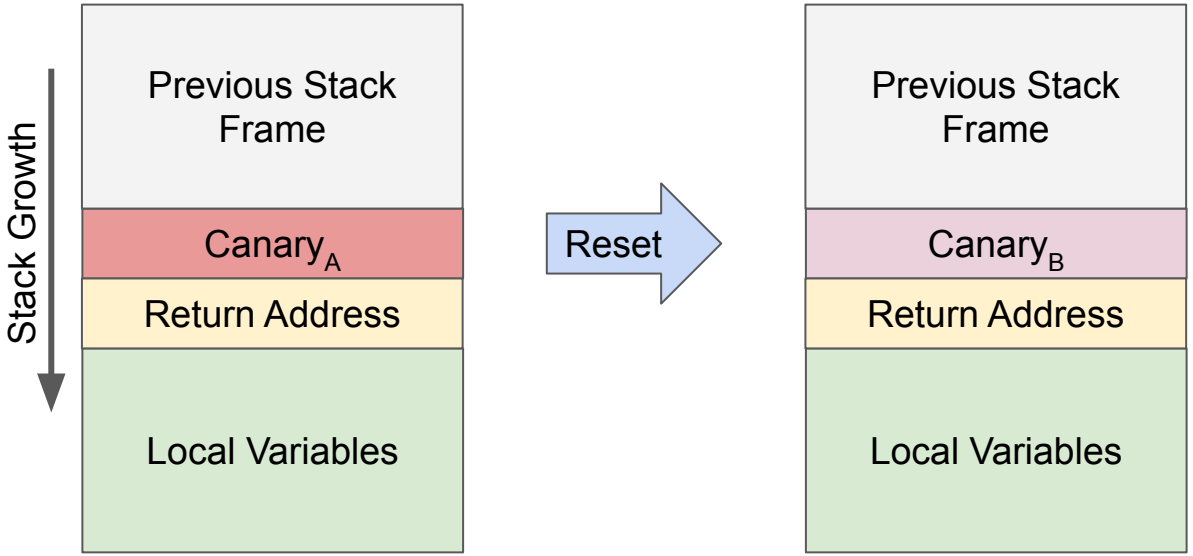
- Randomized Stack Canaries



Case Study - **Flight Controller**

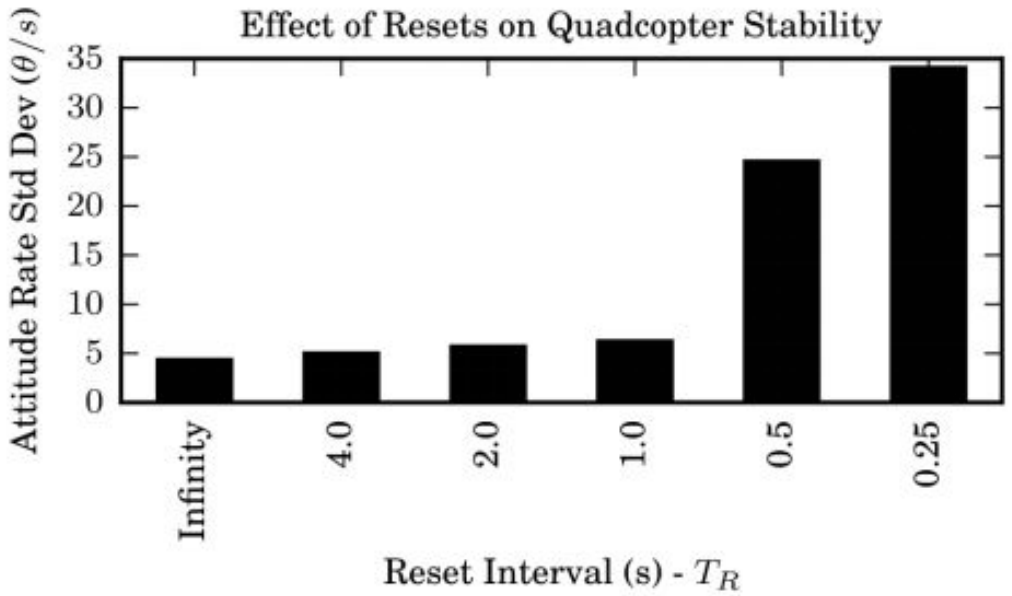
Diversify Strategy

- Randomized Stack Canaries



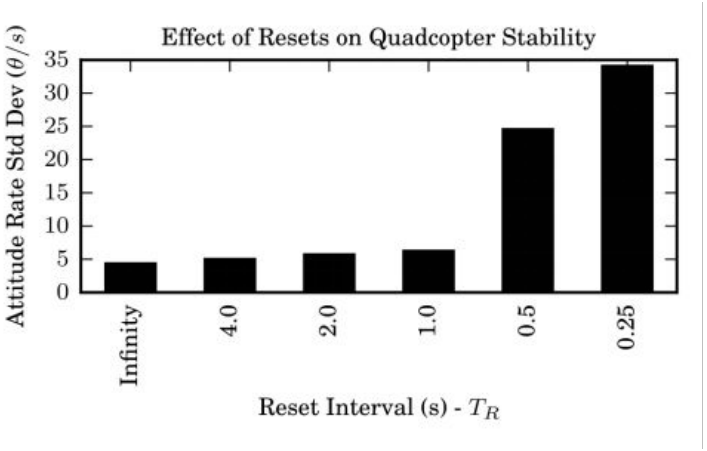
Case Study - **Flight Controller**

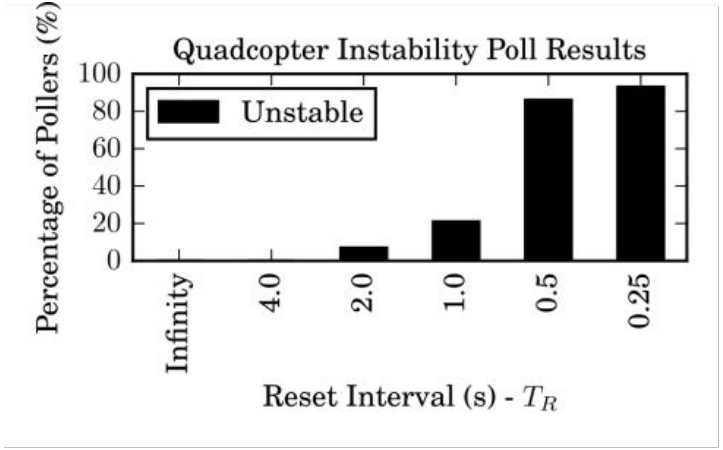
YOLO Performance



Case Study - **Flight Controller**

YOLO Performance





Summary

- CPS properties can strengthen security.
- Eliminates malware from a system (RESET step).
- Increased work for an attacker (DIVERSIFY step).

Summary

- CPS properties can strengthen security.
- Eliminates malware from a system (RESET step).
- Increased work for an attacker (DIVERSIFY step).

Questions?

Intentionally Left Blank

YOLO: Limitations & Mitigations

- Multiple Interacting Components
  - Timing and communications challenges may be mitigated by a microreboot like approach [2].
- Temporary loss of control
  - Replication & Interleaved resets can help alleviate this issue.
- Orthogonal Concerns
  - Spoofed inputs, algorithm stability, etc solutions can be layered with YOLO.