MICHIGAN STATE
UNIVERSITY

Project Plan Presentation
Android Exploit Fuzzing Analysis

The Capstone Experience

Team Google

Karan Singh
Romario Rranza
Shubham Chandna
Anurag Kompalli
Michael Umanskiy
Catherine Xu

Department of Computer Science and Engineering
Michigan State University

Fall 2022

From Students…
…to Professionals

Project Sponsor Overview

- Google – Tech
  - Founded: Menlo Park, CA
    - Detroit, MI; Seattle, WA
    - 50 Countries; 70 Offices
  - Main Product: Search Engine
  - Revenue Source: Ad services
  - Internet connectivity; Smart devices
    - Google Chrome, Google Home
  - Developer of Android OS

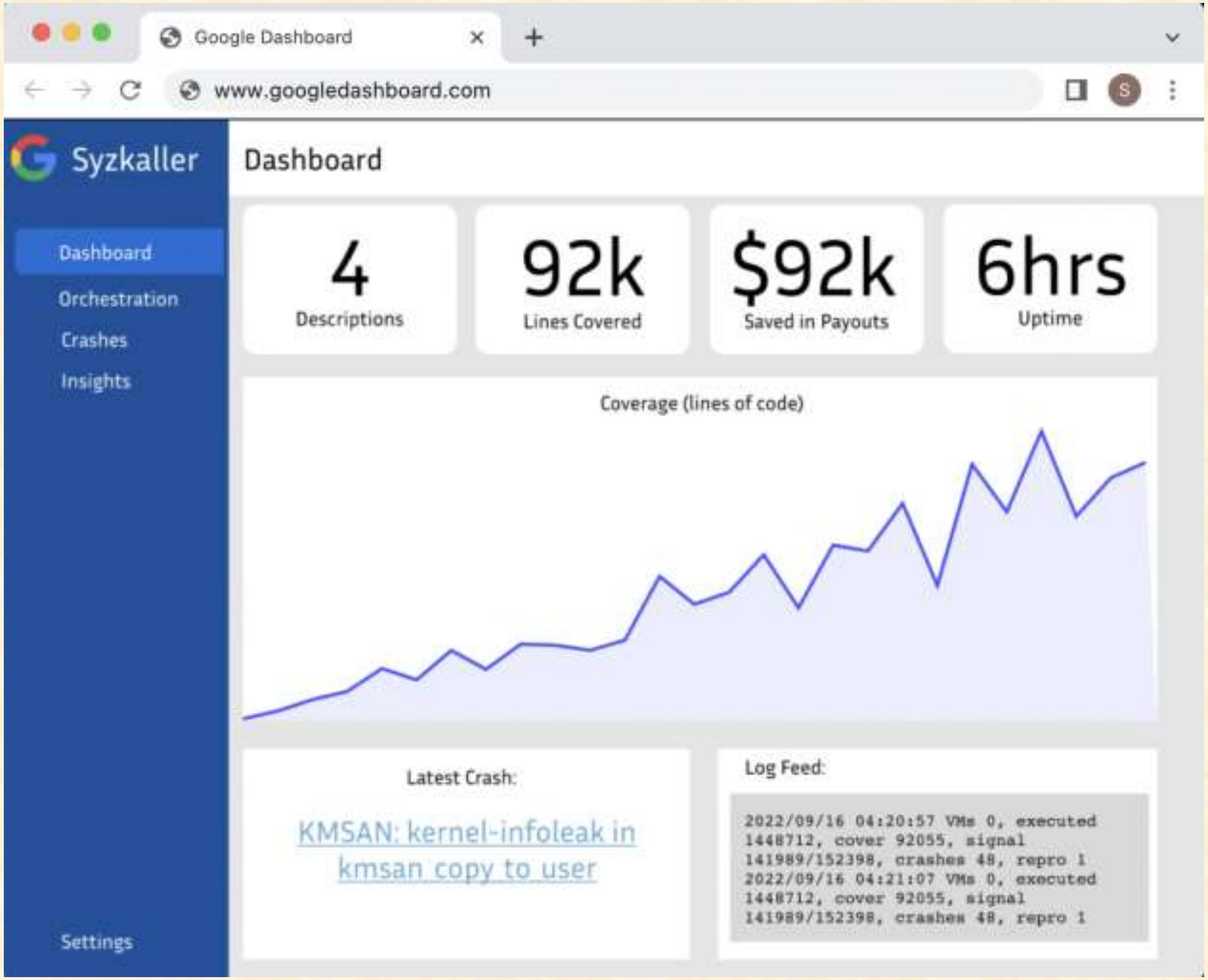
Project Functional Specifications

- Find bugs in existing Android software
- Display bugs in an intuitive manner on a dashboard
  - Allow for a more in-depth look at any bugs found using the fuzzer
- Control Fuzzer Instances from the Dashboard

Project Design Specifications

- Dashboard Tab
  - Gives a “snapshot” of the fuzzer at that time
- Orchestration
  - Start and stop fuzzer instances on the fly
  - Allows for custom configurations
- Crashes
  - A peek into where the fuzzer detected unusual behavior
- Insights
  - Visualizations to provide a better view into the fuzzer metrics

Screen Mockup: Dashboard



Dashboard

4 Descriptions

92k Lines Covered

$92k Saved in Payouts

6hrs Uptime

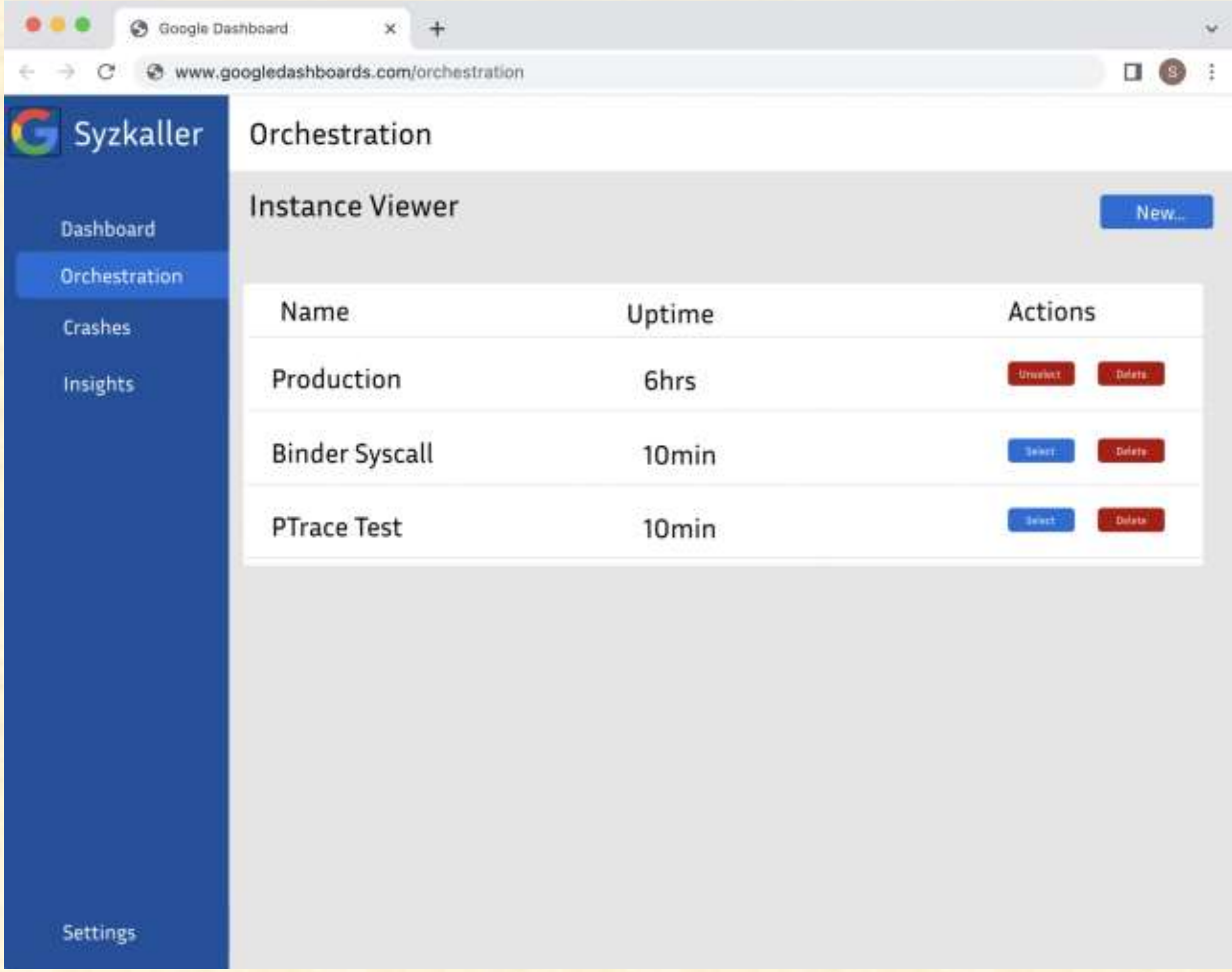
Latest Crash:

KMSAN: kernel-infoleak in kmsan copy to user

Log Feed:

```
2022/09/16 04:20:57 VMs 0, executed
1448712, cover 92055, signal
141989/152398, crashes 48, repro 1
2022/09/16 04:21:07 VMs 0, executed
1448712, cover 92055, signal
141989/152398, crashes 48, repro 1
```

Screen Mockup: Orchestration

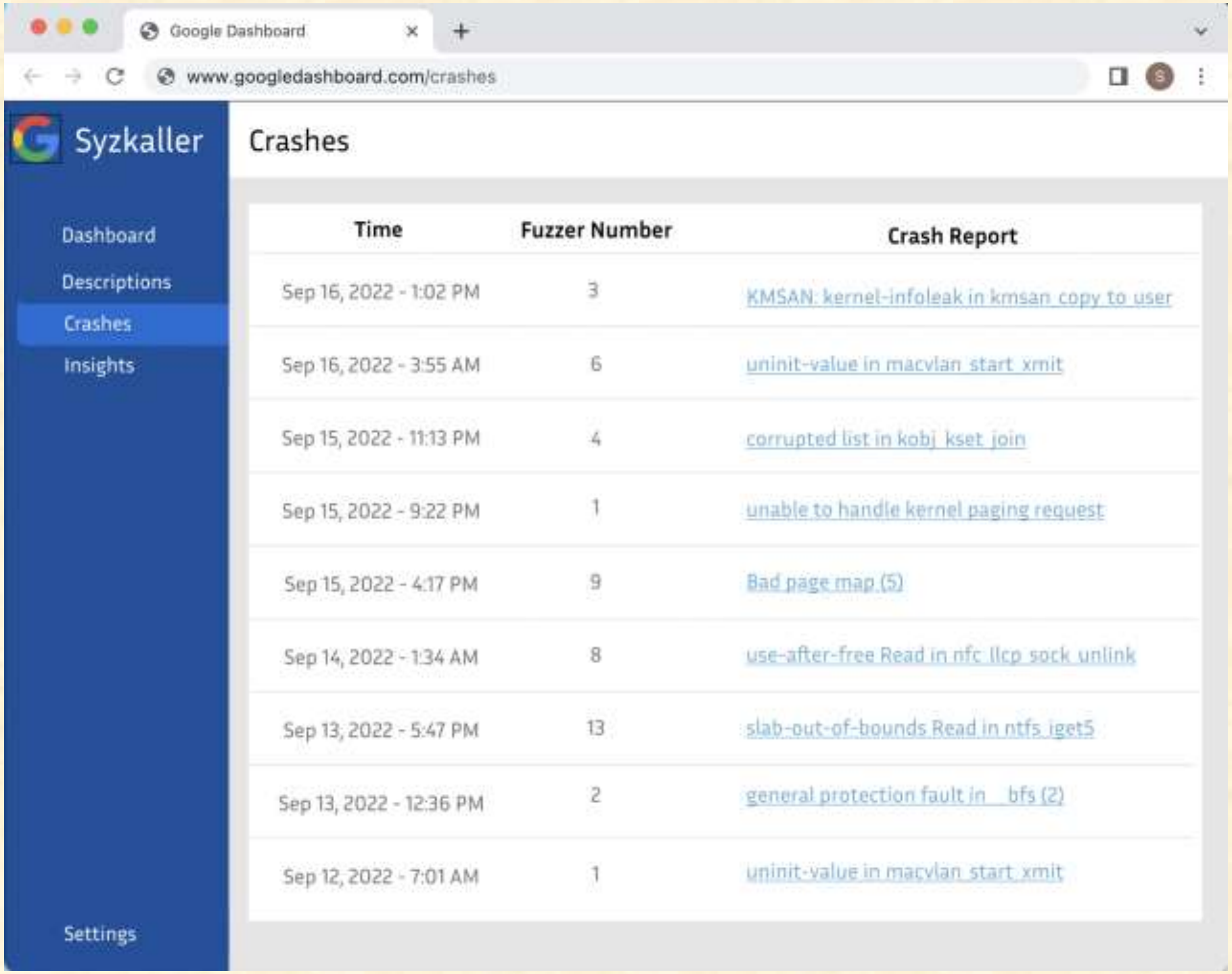


Orchestration

Instance Viewer

Name	Uptime	Actions
Production	6hrs	Unselect Delete
Binder Syscall	10min	Select Delete
PTrace Test	10min	Select Delete

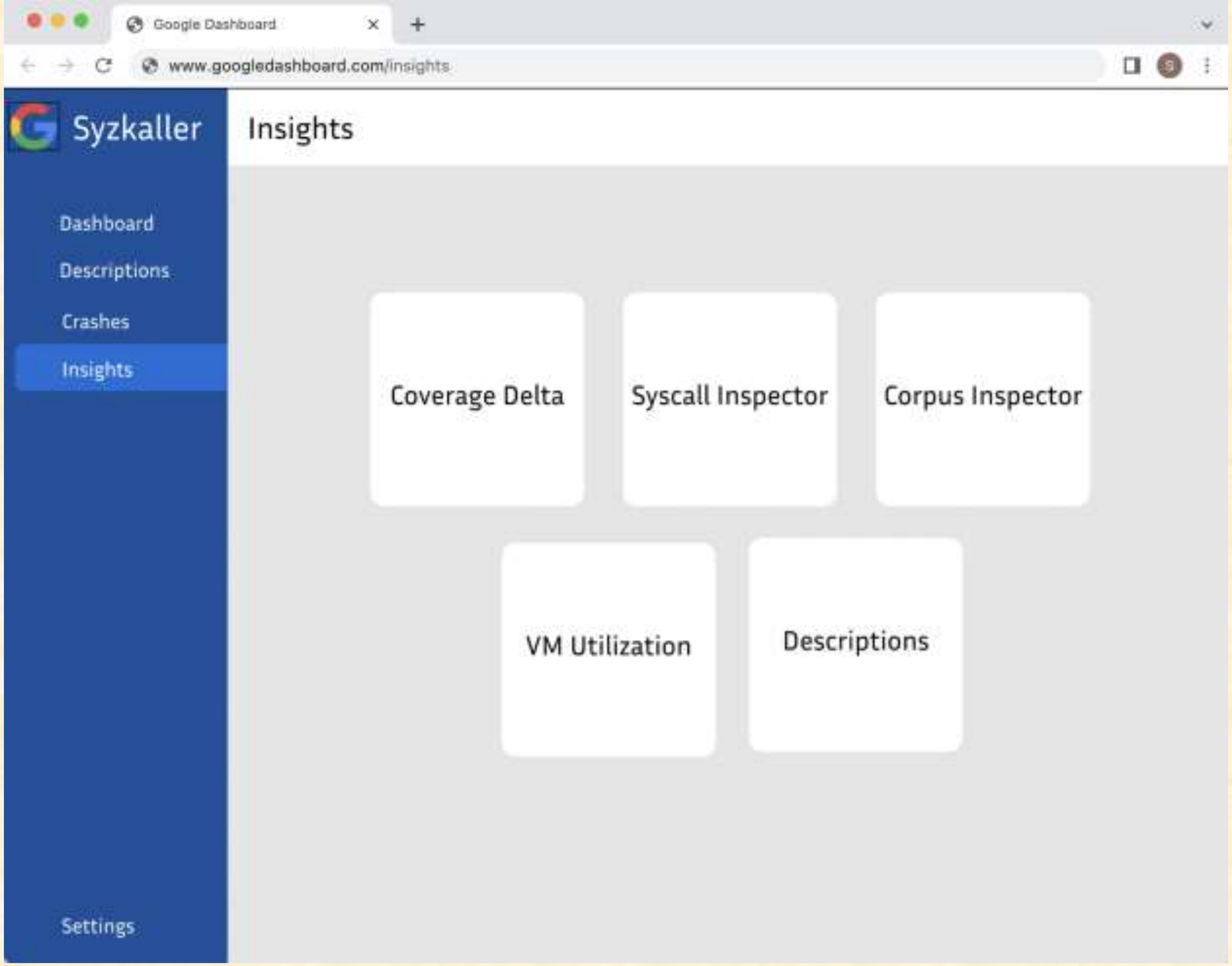
Screen Mockup: Crashes



Crashes

Time	Fuzzer Number	Crash Report
Sep 16, 2022 - 1:02 PM	3	KMSAN: kernel-infoleak in kmsan copy to user
Sep 16, 2022 - 3:55 AM	6	uninit-value in macvlan start xmit
Sep 15, 2022 - 11:13 PM	4	corrupted list in kobj kset join
Sep 15, 2022 - 9:22 PM	1	unable to handle kernel paging request
Sep 15, 2022 - 4:17 PM	9	Bad page map (5)
Sep 14, 2022 - 1:34 AM	8	use-after-free Read in nfc llcp sock unlink
Sep 13, 2022 - 5:47 PM	13	slab-out-of-bounds Read in ntfs iget5
Sep 13, 2022 - 12:36 PM	2	general protection fault in bfs (2)
Sep 12, 2022 - 7:01 AM	1	uninit-value in macvlan start xmit

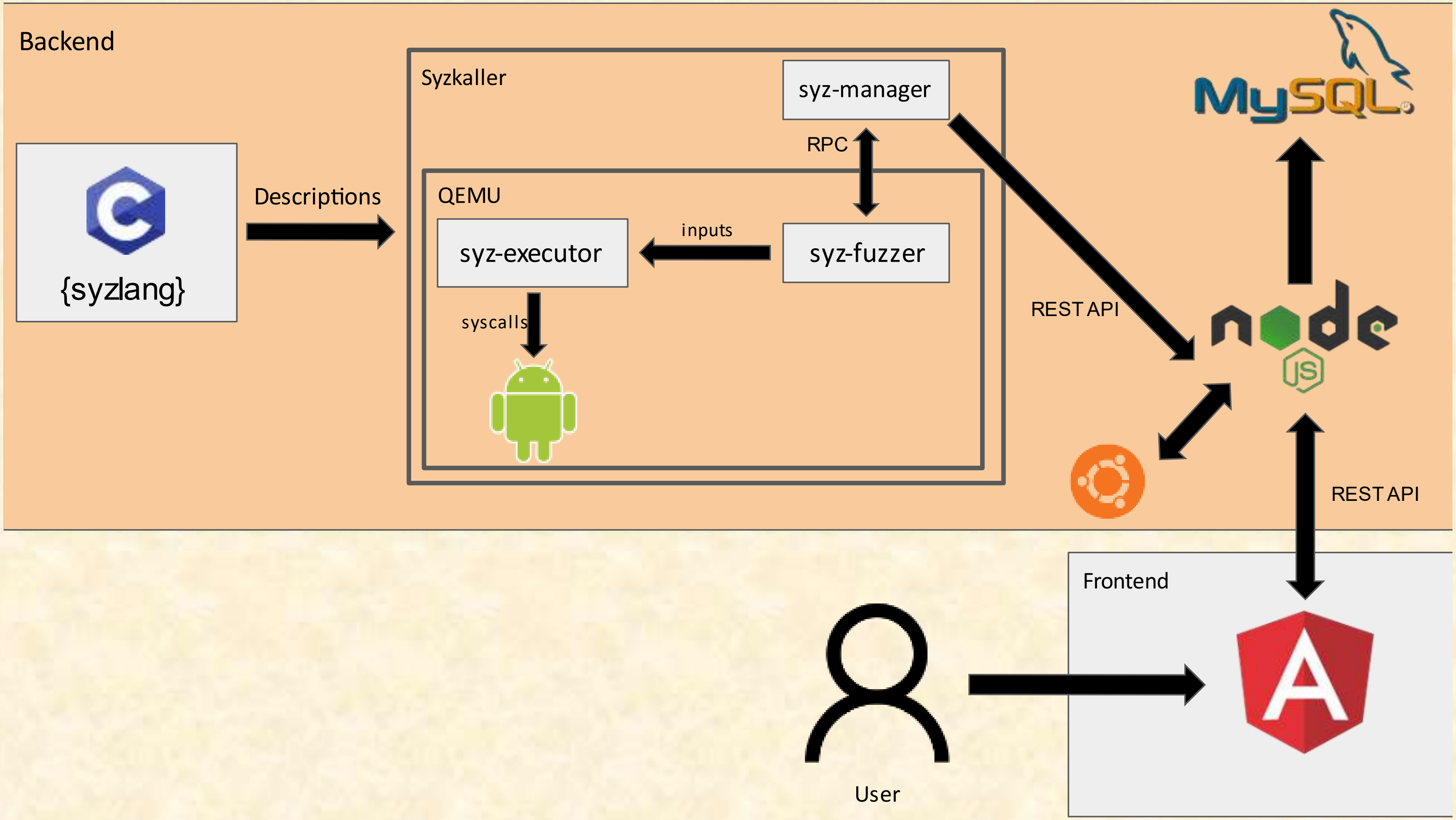
Screen Mockup: Insights



Project Technical Specifications

- Use syzlang to write syzkaller descriptions to “fuzz” the Android kernel for bugs
- Syz-manager orchestrates all Linux Kernel VMs to fuzz on.
- Node.JS used to start and stop Node.JS instances.
- Angular dashboard hits Node.JS API for data generated by syzkaller

Project System Architecture



Project System Components

- Hardware Platforms
  - Rack Mounted Server
- Software Platforms / Technologies
  - Ubuntu
  - Android VMs for Syzkaller
  - Angular
  - NodeJS
  - Syzkaller
  - QEMU
  - MySQL

Project Risks

- Getting and Computing Metrics from Syzkaller [Medium]
  - We need to be able to pull metrics out of Syzkaller in an easy-to-use way, such as JSON.
  - Modify syzkaller code to expose an API endpoint that returns the data in JSON rather than HTML so we can more easily work with it.
- Controlling Syzkaller from the Dashboard [Hard]
  - We need to be able to manage the lifecycle of a syzkaller instance from start to stop from the dashboard. This isn't an easy problem to solve due to the environment that syzkaller needs to operate in.
  - Investigate using the "child_process" package for Node.JS to start syzkaller from the shell. Alternatively, we can explore using Docker to start full instances and manage them.
- How to Prioritize, Visualize and Calculate Metrics [Medium]
  - Due to the vast number of ways to visualize data and our inexperience with fuzzing, we are not sure how best to make the insights portion of our application. We are unsure how to prioritize and visualize certain metrics that may be useful for the insights portion of our application.
  - There has been work done at Google for fuzzing data visualization, but it falls on us to flesh out the final product. We can utilize resources that Google gives us and combine them with our gained experience writing fuzzer descriptions to produce insightful visualizations.
- Figuring out where Descriptions are Incomplete [Medium]
  - Since the syzkaller tool is mature, many descriptions already exist. A challenge for us will be finding out how this system is lacking despite our inexperience with kernel development
  - Using our sponsors knowledge with the Linux Kernel, we can get guidance on which *areas* might be incomplete, which will ease our search process for areas to contribute.

Questions?

? ? ? ?

? ?

? ? ?