VALIDATION AND VERIFICATION
TP 1

Aymeric Pierre

Mathieu Merien

1 Article news about a software bug

The high voltage management system alarm had a race condition bug because 2 processes were able to access a data structure at the same time, causing the system to silently fail. This caused the system to get stuck in an infinite loop. The bug by allowing an accumulation of unprocessed events also causes the backup system to fail. This is a global bug, as it is triggered by a unique combination of events The lack of alarm and the accumulation of unprocessed events revealed the bug. The effect was the great blackout of 2004 [1], [2]. To find the bug, a way to test the software can be the following:

- First, create a test suite that tests all the basic functionality,
- then switch between running the test suite and sending a batch of random events.

The bug could be detected

2 Analysis of Apache bug

A dictonary is transformed into a string. Previously it was assumed that the order of the different elements in the dictonary would have a fixed order in the string. But it wasn't, and the author took this possible case into account in the code.

old code:

```
assertEquals("{A=[X, Y, Z], B=[U, V, W]}", map.toString());
```

new code:

```
assertTrue(
            "{A=[X, Y, Z], B=[U, V, W]}".equals(map.toString()) ||
            "{B=[U, V, W], A=[X, Y, Z]}".equals(map.toString())
);
```

Code coverage didn't change after they fixed it, so they didn't add new tests. [3]

3 Chaos Engineering

Chaos Engineering is a practice used to test big distributed system with failure experiments. Because the complexity of Netflix's infrastructure made the traditional method of testing impossible, they invent a new way to test it with some experiments done in production.

For each experiment, it is necessary to measure the normal state of the system and hypothesize that this measure will remain the same after the failure is introduced. Then you introduce the failure and try to see, with the measurement, if the system can handle this kind of failure. For Netflix, the variable that indicates the normal state of the system is the number of streams started per second.

These failures can be :

- Terminate the virtual machine instance
- latency in requests between services
- Fail inter-service request
- Failure of an internal service
- make an entire Amazon region unavailable

According to the paper, Netflix isn't the only company conducting these experiments; all the major companies, including Amazon, Facebook and Microsoft, are doing the same. [4]

4 Web Assembly (WASM)

A formalised semantics allows many advantages. First, by being hardware, platform, language and browser independent, it allows better portability. It also allow improvement of speed and security. In last it also allow determinism.

Your formalise semantics has allowed to prove the absence of type and memory bugs, but is not the proof that the web assembly doesn't have any of them.

5 Mechanized Specification in Web Assembly

The project aims to create a mechanized specification of the WASM language, its main benefit being a verified interpreter and full proof type checker. The project comes with several proofs of the soundness of the type checker, and found several defects in the official WASM specification that need to be fixed. This mechanized specification creates artifacts with external third party programs required to run the code, which unfortunately create an untrusted interface. This specification has been validated by running the full official WASM test suite and some other experiments. During the validation, the teams found a flaw in some commercial specifications. This project doesn't eliminate the need for testing, as only the type checking system has been fully tested. But it creates a more truthful specification of WASM.

Bibliography

[1] K. Poulsen, "Software Bug Contributed to Blackout," *SECURITYFOCUS NEWS*, 2004, [Online]. Available: https://web.archive.org/web/20040313012800/http://www.securityfocus.com/news/8016

[2] K. Poulsen, "Tracking the Blackout bug," *theregister*, 2004, [Online]. Available: https://www.theregister.com/2004/04/08/blackout_bug_report/

[3] B. P. Kinoshita, "AbstractMultiValuedMapTest#testToString is flaky," 2020, [Online]. Available: https://issues.apache.org/jira/projects/COLLECTIONS/issues/COLLECTIONS-771?filter=doneissues

[4] A. Basiri *et al.*, "Chaos Engineering," *IEEE Software*, vol. 33, no. 3, pp. 35–41, May 2016, doi: 10.1109/MS.2016.60.