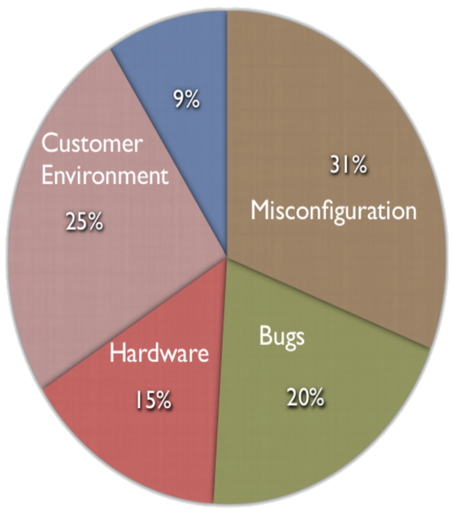
Version Space Learning for Verification on Temporal Differentials

Mark Santolucito
Ennan Zhai, Ruzica Piskac
Yale University

Motivation

- Software or service failure are very expensive.
- Software misconfiguration problems are the most common root-cause (31%), e.g., Amazon EC2 outage Apr 2011.



DATA CENTER KNOWLEDGE

Amazon: Networking Error Caused Cloud Outage

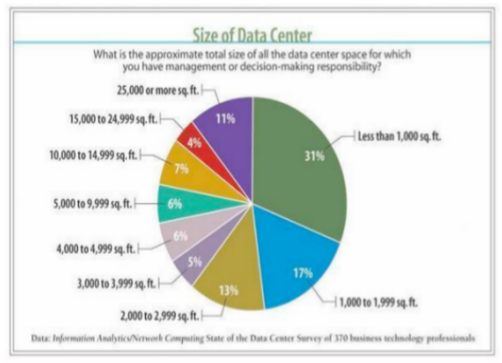
By: Rich Miller
April 29th, 2011

Like 19 Tweet +1 Share Print

Last week's lengthy outage for the Amazon Web Services cloud computing platform was caused by a network configuration error as Amazon was attempting to upgrade capacity on its network. That error triggered a sequence of events that culminated in a "re-mirroring storm" in which automated replication of storage volumes maxed out the capacity of Amazon's servers in a portion of their platform.

Service Outages Generate Big Losses

Downtime of cloud service can cost an average of $505,500 per incident, according to a Ponemon Institute study.



Analytics Slideshow: 2010 Data Center Operational Trends Report

Misconfiguration examples

```
extension = mysql.so
  … …
extension = recode.so
```

"recode.so" must be put before "mysql.so"

Problem Type: Ordering error

Description: When using PHP in Apache, the extension "**mysql.so**" depends on **"recode.so**". Thus, the order between them matters. The user configured the order in a wrong way.

Impact: Apache cannot start due to segment fault

```
general_log = /var/log/mysql/mysql.log
```

Problem Type: Value type error

Description: The parameter "general_log" should be an integer, rather than path (string). In MySQL, there is another parameter "general_log_file" used to point the log path.

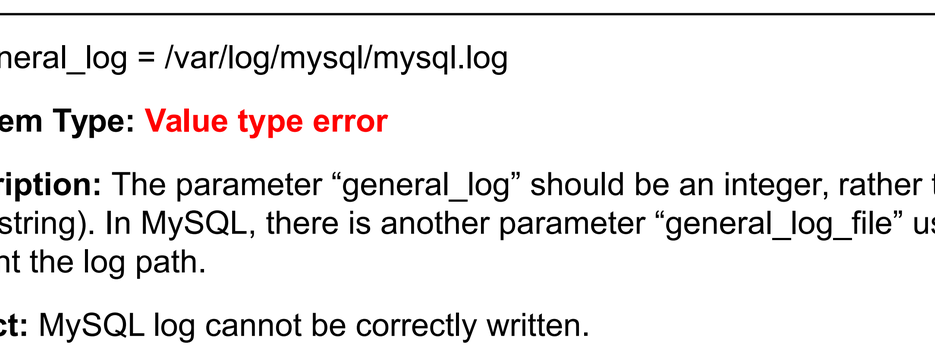
Impact: MySQL log cannot be correctly written.

Solution Attempt #1

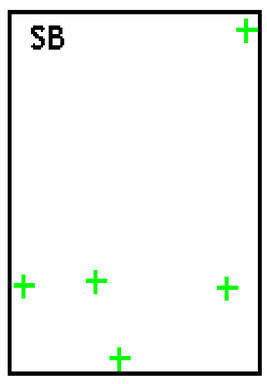
- In our previous work, **ConfigC**[1] can learn a language model from a learning set of configuration files by building a set of necessary relations over a learning set of correct files.

File F
A=1
B=2
C=3

Relations (F) =
{ A<B, B<C, A<C,
(A,B), (B,C), (A,C),
A:Int, B:Int, C:Int, …}



- ConfigC is an instance of **version space learning**. But it only builds the specific boundary (Necessary set) – the strongest conditions for a correct file.

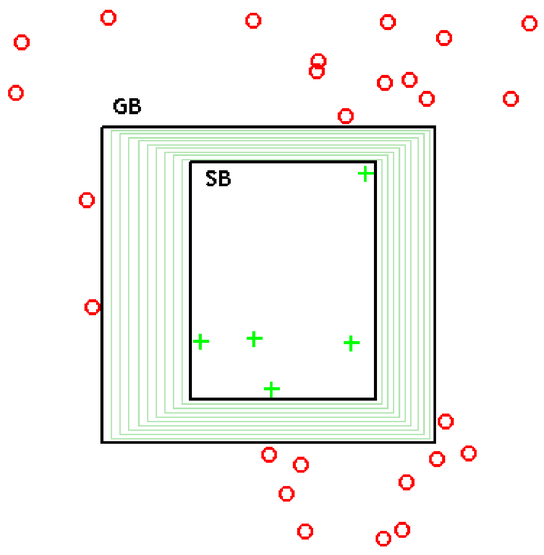


Error Type	Relations	Passing Tests	False Positives
Missing Entry	X in same files as Y	5/5	1, 0, 0, 0, 4
Type Error	X : Int	5/5	0, 0, 0, 0, 0
Keyword Ordering	X before Y	5/5	0, 2, 1, 0, 6
Value Relations	X > Y, X=Y	4/5	0, 0, 0, 1, 0

- This is guaranteed to detect all incorrect files, but also generates many false positives – errors that are not true errors.

Solution Attempt #2

- We extend ConfigC to also build the general boundary – the set of weakest condition for a correct file.

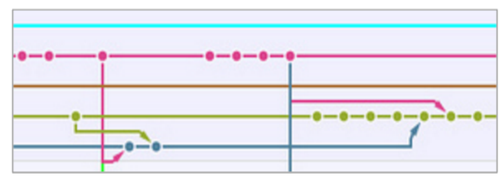


- Instead of building a concrete relation set from the learning files, we build an SMT formula in the theory of sets.

Greatest Boundary = Breaking : {Relations}
Specific Boundary = Necessary : {Relations}

Status(F) = Pass ⇒ ∀r ∈ Relations (F), r ∉Breaking
Status(F) = Err ⇒ ∃r ∈ Relations (F), r ∈Breaking

- The formula can be extended with extra observations - for example using temporal structures in the learning set.



Status(F1) = Pass ∧ Status(F2) = Err ⇒
∃r ∈ Relations (F1) `setDiff` Relations (F2), r ∈ Necessary ⋁
∃r ∈ Relations (F2) `setDiff` Relations (F1), r ∈ Breaking

Proposed Application

- Travis Continuous Integration[2] service for testing
- ~30% of large projects on Github use TravisCI
- 15-20% of failed TravisCI builds are due to "errors"
- Since the start of 2014, approximately 88,000 hours of server time was used on TravisCI projects that resulted in an error status.
- Because commits are incremental, set differences are small, which lets the SMT solver run relatively quickly.

Relations (F1) `setDiff` Relations (F2) << Relations (F2)

[1] M. Santolucito, E. Zhai, and R. Piskac, "Probabilistic automated language learning for configuration files," in CAV, 2016, pp. 80–87.

[2] Z. A. Beller M, Gousios G, "Oops, my tests broke the build: An analysis of travis ci builds with github," PREPRINT, 2016. [Online]. Available: https://doi.org/10.7287/peerj.preprints.1984v1

This work was funded in part by NSF grant #1302327