Data-Driven Device Failure Prediction

Working Title

Paul L. Jordan

Air Force Institute of Technology
Center for Cyberspace Research

18 August 2016

Introduction

- Problem
- Solution
- Progress
- Results
- Future Work

Problem

- Predict failure given indicators in existing log messages
  - Survey paper on machine learning techniques for doing this [4]
- Need labelled training data
  - Adaptive Failure Prediction (AFP) framework [1]
  - AFP wasn't capable of running on modern operating system
  - AFP didn't exhaustively emulate all possible/realistic faults [3]

Solution

- Implement and modernize AFP with more representative fault load
    - Need realistic workload generator
    - Need to modernize and adapt fault injection tool
    - Need to design ways of emulating more realistic fault-load

Adaptive Failure Prediction Framework

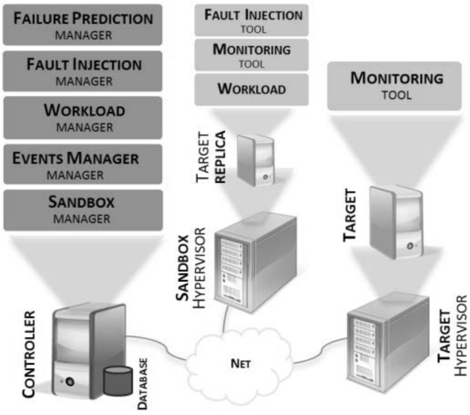


Figure: The Adaptive Failure Prediction (AFP) framework [1].

Progress

- ✓ Virtual environment implemented
- ✓ Load generator implemented
- ✓ Fault-injection tool implemented
- ✓ Experiments complete

Results

- Three additional faults tested
  - Under-resourced CPU
  - Third-party application memory leak
  - Third-party memory corruption

Results

- Fault-Injection
  - Target process crashes immediately
  - No indicators to use to train machine learning algorithm

Results

- Under-Resourced CPU
  - Response times were drastically increased
  - Target process would not fail

Results

- Memory Corruption
  - Different from fault-injection in that it corrupts heap-space instead of program memory
  - Same as fault-injection: either wouldn't fail, or would crash immediately with no warning signs

Results

- Memory Leak
  - Only fault load that caused failure with indicators present in log messages prior to failure
  - Trained two statistical models (Support Vector Machine, and Boosted Decision Tree)
  - As expected, both predictors performed adequately before software update, then poorly after
  - After re-training with newly generated data performance once again was adequate

Results

- What is adequate?
  - Naïve predictor predicts non-failure prone at all times
  - Currently no form of prediction is taking place in operational environment
  - Machine learning classification algorithms evaluated using ROC and Precision/Recall Curves [4, 2]

Results

Sample ROC and Precision Recall Curves:

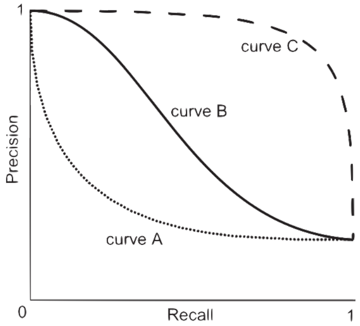


Figure: Sample precision/recall curves [4]. Curve *A* represents a poorly performing predictor, curve *B* an average predictor, and curve *C* an exceptional predictor.

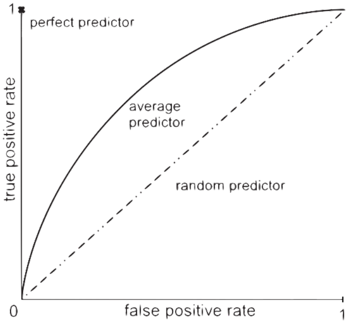
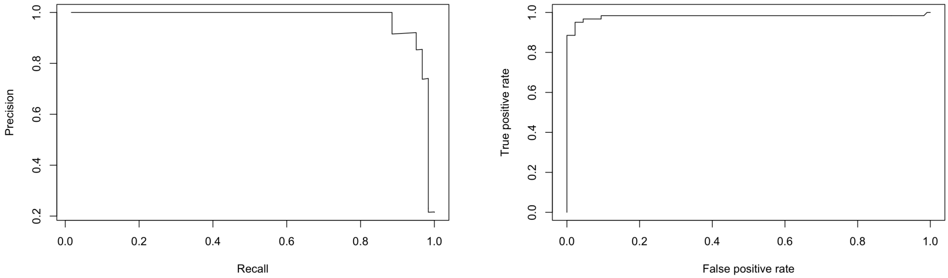


Figure: ROC plots of perfect, average, and random predictors [4].

Results

- Boosted Decision Tree Performance



(a) Precision/Recall Curve.

(b) ROC Curve (AUC = 0.9801).

Figure: Performance of the boosting prediction method trained on failure data created after the software update obtained by consuming all available memory until target application fails.

Future Work

- Further validation and automation
- Implement and make operational
- Further explore fault injection

Summary

- Domain Controller (*lsass.exe*) is relatively robust process
- Unmodified, AFP incompatible with modern domain controller
- Extended AFP capable of automatically training an effective failure prediction model
- Extended AFP is able to adapt to underlying system changes to minimize impact on manpower

Questions?

I. Irrera, M. Vieira, and J. Duraes.
Adaptive failure prediction for computer systems: A framework and a case study.
In *Proceedings of the 2015 IEEE 16th International Symposium on High Assurance Systems Engineering (HASE 2015)*, pages 142–149, 2015.

G. James, D. Witten, T. Hastie, and R. Tibshirani.
An Introduction to Statistical Learning: With Applications in R.
Springer Publishing Company, Incorporated, 2014.

N. Kikuchi, T. Yoshimura, R. Sakuma, and K. Kono.
Do injected faults cause real failures? a case study of linux.
In *Proceedings of the 25th IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW 2014)*, pages 174–179, 2014.

F. Salfner, M. Lenk, and M. Malek.
A survey of online failure prediction methods.
ACM Computing Surveys (CSUR), 42(3), 2010.