

# Curriculum Vitae: Karthik Pattabiraman

---

**Address:** Fred Kaiser Building, Univ. of British Columbia, 2332 Main Mall, Vancouver, BC Canada V6T 1Z4.

**Email:** [karthikp@ece.ubc.ca](mailto:karthikp@ece.ubc.ca)

**Alternate Email:** [Karthik.Pattabiraman@gmail.com](mailto:Karthik.Pattabiraman@gmail.com)

**Phone:** (604)-827-4245 (Office)

**Webpage:** <http://blogs.ubc.ca/karthik/>

## Research Interests

Error Resilient Systems, Compilers and Programming Languages, Web Applications' Reliability, IoT Security.

## Education

PhD in Computer Science	University of Illinois (Urbana Champaign) <i>Advisor: Ravishankar K. Iyer</i>	May 2009
MS in Computer Science	University of Illinois (Urbana Champaign) <i>Advisor: Daniel A. Reed</i>	Dec 2004
B.Tech in Information Technology	University of Madras (Chennai, India)	Jul 2001

## Awards and Honors

- My student Guanpeng Li received Standard Performance Evaluation Corporation (SPEC) Kaivalya Dixit Distinguished Dissertation award in 2019. This award “aims to recognize outstanding doctoral dissertations in the field of computer benchmarking, performance evaluation, and experimental system analysis in general”. One award is given each year.
- Paper at ISSRE'14 was chosen as the “Highlights of ISSRE” in ISSRE 2019 – one of 26 papers among 1074 published in 30 years of the IEEE International Symposium on Software Reliability Engineering (ISSRE). These are papers that have had “great influence and impact in the community”.
- UBC Killam Faculty Research Prize, 2018 in the Sciences and Engineering category “in recognition of outstanding research and scholarly contributions” (only prize awarded in the “Junior” category – less than 12 years since PhD).
- University of Illinois's (UIUC) Computer Science Department Distinguished Alumni Award – Early Career Educator Category (one of two alumni awarded in this category), 2018.
- Best Paper Award Nominee at the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2018. One of three papers that were nominated out of nearly 250 submissions.
- Distinguished paper awards at the European Dependable Computing Conference (EDCC), 2015 and 2016 (twice). One of three papers from more than 50 submissions in each year.
- Killam Research Fellowship at UBC for 2016. This fellowship is awarded on a competitive basis to 10 researchers each year across all of UBC and all ranks (for salary and research supplement during a study leave, i.e., sabbatical).
- NSERC Discovery Accelerator Grant Supplement (DAS) Award for 2015-2018. “The DAS Program provides substantial and timely resources to researchers who have a superior research program that is highly rated in terms of originality and innovation, and who show strong potential to become international leaders within their field.” One of 125 awarded across all fields of science and engineering out of more than 3000 applications submitted in 2015 in Canada.
- ACM SIGSOFT Distinguished paper award at the IEEE/ACM International Conference on Software Engineering (ICSE) 2014, one of nine papers that received the award (out of nearly 500 submissions).
- Best paper award runner up at the IEEE International Conference on Software Testing, Verification and Validation (ICST) 2013. (one of 2 papers among more than 150 submissions).
- Best paper award nominee at the IEEE International Conference on Software Testing, Verification and Validation (ICST) 2012. (one of 6 papers among more than 150 submissions).
- Nominated by the University of Illinois's Computer Science Department for the 2009 *Ross J. Martin* award, given to one student in the college of engineering based on research excellence. Each department is allowed one nominee
- Winner of the 2008 *William C. Carter* award sponsored by the IEEE Technical Committee on Fault-Tolerant Computing (TC-FTC) and the IFIP Working Group on Dependable Computing and Fault Tolerance (WG 10.4). The William C. Carter Award is presented annually since 1997 “to recognize an individual who has made a significant contribution to

# Curriculum Vitae: Karthik Pattabiraman

---

*the field of dependable computing through his or her graduate dissertation research*". The award is given to one student a year at the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN).

## Work Experience (post PhD)

- Associate Professor, Dept. of Electrical and Computer Engineering, Univ. of British Columbia, July 2015-present.
- Assistant Professor, Dept. of Electrical and Computer Engineering, Univ. of British Columbia, Jan 2010- Jun 2015.
- Post-Doctoral Researcher, Microsoft Research, Jan-Dec 2009, Research in Software Engineering (RiSE) Group

## Current Graduate Students (PhD/MASc)

1. Mohammad Rafiuzzaman, PhD, from 2017. (co-advised with Sathish Gopalakrishnan). International Doctoral Fellowship (IDF) award.
2. Kumseok Jung, PhD, from 2017. Interned at Microsoft Research.
3. Mehdi Karimi, PhD, from 2017 (co-advised with Andre Ivanov). Four Year Fellowship (FYF) awardee.
4. Asem Ghaleb, PhD, from 2018. Four Year Fellowship (FYF) awardee.
5. Aarti Kashyap, MASc, from 2018.
6. Abdul Rehman Anwer, MASc, from 2018.
7. Zitao Chen, MASc, from 2018.
8. Pritam Dash, MASc, from 2018.
9. Niranjhana Narayanan, MASc, from 2019.

**Visiting Student:** Arpan Gujarati, PhD student, MPI-SWS, Germany (joint with Sathish Gopalakrishnan).

## Former Graduate Students and Post-docs

### Post-doc

1. Julien Gascon-Samson. NSERC Post-Doctoral Fellow, 2016-2018. Currently assistant professor at ETS Montreal.

### PhDs:

1. Bo Fang, PhD, from 2014, MASc from 2011-2014 (co-advised with Matei Ripeanu). Interned at Los Alamos National Labs (LANL), Pacific Northwestern National Labs (PNNL). J.K. Zee Fellowship. (received the NSERC Post-doctoral Fellowship in 2020 - ranked #2 in CS). First Position: Post-doctoral researcher at Pacific Northwestern National Labs.
2. Guanpeng Li, PhD, from 2019. Four Year Fellowship (FYF) awardee. Interned at Nvidia Research. Distinguished paper runner up at DSN 2018. Co-authored papers: [2][22][23][32][26][38][42] [16]. (received the NSERC Post-doctoral Fellowship in 2019). Joining University of Iowa (U Iowa), Computer Science department soon as assistant professor. First Position: Post-doctoral researcher, University of Illinois (UIUC). Winner of the **SPEC Kaivalya Dixit Award**, 2019.
3. Farid Molazem Tabrizi, PhD, 2017. Interned at Fortinet, Vancouver, and Google, Mountain View. Distinguished paper award at EDCC 2015. Co-authored papers: [31][40][54]. First Position: Google, US.
4. Saba Alimadadi, PhD, 2017 (co-advised with Ali Mesbah), Distinguished paper award at ICSE 2014. Interned at SAP Vancouver. Co-authored Papers: [5][37][41][51]. Joining Simon Fraser University (SFU), Computer Science department soon as assistant professor. First Position: Post-doctoral researcher at Northeastern University, US (received the NSERC Post-doctoral Fellowship in 2018 – ranked #1 in CS).
5. Frolin Ocariza, PhD, 2016 (co-advised with Ali Mesbah), MASc, 2012: NSERC CGS(D) and FYF awardee. Co-authored papers [3][7][44][50] [56][61][62] Interned at Microsoft Research, Redmond, and Fujitsu Labs, America. First position: SAP, Vancouver.
6. Shabnam Mirshokraie, PhD, 2015 (co-advised with Ali Mesbah), Best paper runner up at ICST 2013. Co-authored papers [55][58] [43][8][15]. First Position: Co-founder of a Startup in Stealth mode.

# Curriculum Vitae: Karthik Pattabiraman

---

7. Layali Rashid, PhD, 2013 (co-advised with Sathish Gopalakrishnan), NSERC CGS(D) and FYF awardee. Co-authored papers [8][48][59][65][97][102][103][104]. Interned at Microsoft Research, Redmond. First position: Qualcomm, US.

## Masters:

1. Jiesheng Wei, MASc, 2012. Co-authored papers [49][60][96][98][102][90]. First position: Microsoft Canada.
2. Anna Thomas, MASc, 2013: Co-authored papers [49][57][96][90]. First position: IBM Canada.
3. Majid Dadashi, MASc, 2014: Co-authored papers [48][97]. First position: 1-QBit.
4. Xin Chen, MASc, 2014. Co-authored papers [45] [95]. First position: CCB Fintech, China.
5. Sheldon Sequira, MASc, 2014 (co-advised with Ali Mesbah), Distinguished paper award at ICSE'14. Co-authored papers [51]. First Position: SAP, Vancouver.
6. Kartik Bajaj, MASc, 2014 (co-advised with Ali Mesbah), FYF awardee, MITACS Globalink Fellowship and FYF awardee. Vancouver. Co-authored papers [39][46] [52][56]. First Position: Hootsuite, Vancouver.
7. Qining Lu, MASc, 2015. Co-authored papers [47][42][90]. First position: Google, Canada.
8. Nithya Narayanamurthy, MASc, 2015. Co-authored paper [15]. Distinguished paper award at EDCC 2016. First Position: Oracle Labs, Vancouver.
9. Abraham Chan, MASc, 2017. Co-authored paper [30]. First Position: Huawei, Canada.
10. Maryam Raiyat Aliabadi, MASc, 2018. Co-authored papers [27][34]. First Position: PhD student, Sharif Univ.
11. Ekta Aggarwal, MASc, 2019 (co-advised with Andre Ivanov). Co-authored papers [20][82]. First position: Clio
12. Lucas Palazzi, MASc, 2019. Co-authored papers [16][1]. First Position: Data Realm Inc.

## Visiting PhD students

1. Behrooz Sangchoolie, Chalmers University of Technology, Gothenburg, Sweden (May-August 2016). Co-authored paper: [29]. Current Position: Researcher, Research Institutes of Sweden (RiSE).
2. Yong Yang, PhD student at Peking University, China. (February – August 2019).

**Undergraduate students:** I have mentored a total of 30 undergraduate students as summer interns or undergraduate project advisers at UBCs. Three of these students were NSERC USRA interns (equivalent of NSF REUs in Canada), one was an International USRA intern, and many of them were supported by MITACS Globalink internships.

## Selected Activities

- Steering Committee Member for the IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), 2015-2020. One of ten members of the Steering Committee.
- Guest co-editor of a special issue of the IEEE Transactions on Dependable and Secure Computing (TDSC), 2018 on dependability and security data analysis (one of three co-editors).
- Guest co-editor for a special issue of the IEEE Transactions on Reliability (TR), TBD on Software Reliability Engineering (one of two co-editors).
- Member of the IFIP WG 10.4 on Dependable Computing and Fault Tolerance (January 2015 onwards). Elected vice-chair in 2019 – one of two vice-chairs.
- Senior Member of the IEEE (since 2015), Senior member of the ACM (since 2019), and a member of the Usenix Association (since 2010).
- Co-organizer of a Dagstuhl workshop on “Characterizing and Modeling Residual Software Bugs” (August 2020), one of five co-organizers from different universities.
- Conference Organization
  - Track co-chair for IEEE/ACM Conference on Design Automation and Test in Europe (DATE), 2021 - Dependability and System-Level Test Track (one of two co-chairs)
  - Program coordinator for IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2020.

# Curriculum Vitae: Karthik Pattabiraman

---

- o New Faculty Symposium Co-Chair for IEEE International Conference on Software Reliability Engineering (ISSRE), 2020.
- o Program co-chair for the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2019. (one of two co-chairs)
- o Program co-chair of the second International Workshop on Dependable and Secure Machine Learning (DSML), co-held with the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2019. (one of five co-chairs) – DSML had the largest attendance among all DSN workshops.
- o Program co-chair of the first International Workshop on Dependable and Secure Machine Learning (DSML), co-held with the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2018. (one of three co-chairs)
- o Publications co-chair for the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2018. (one of two co-chairs)
- o Program co-chair of the IEEE Workshop on Silicon Errors in Logic, System Effects (SELSE), 2018. (one of two co-chairs)
- o Program co-chair for the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2017. (one of two co-chairs)
- o Program chair for the 16<sup>th</sup> IEEE International Workshop on Assurance in Distributed Systems and Networks (ADSN), 2017. Held in conjunction with the IEEE International Conference on Distributed Computing Systems (ICDCS), 2017.
- o Member of expert panel for the Special Issue on Software Reliability in Journal of Systems and Software, Journal of Software and Systems (JSS), 2017, based on the best papers from ISSRE 2016.
- o Industry track co-chair for the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2017. (one of two co-chairs)
- o Program co-coordinator: IEEE International Symposium on Software Reliability Engineering (ISSRE), 2016.
- o Workshops co-chair of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2016. (one of two co-chairs)
- o Finance chair of the 14<sup>th</sup> ACM/IFIP/Usenix International Conference on Middleware, 2015.
- o Local Chair of 1<sup>st</sup> IEEE International Conference on Software Quality, Reliability and Security (QRS), 2015.
- o Publicity co-chair of the 11<sup>th</sup> European Dependable Computing Conference (EDCC), 2015.
- o Publications chair of the 8<sup>th</sup> IEEE/ACM International Symposium of Network On Chips (NOCS), 2015
- o Fast abstracts chair, IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2014.
- o Student Track chair, IEEE International Symposium on Software Reliability Engineering (ISSRE), 2014.
- o General chair of the IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), 2013.
- o Program co-chair of the first and second workshops on *Compiler and Architectural Techniques for Application Reliability and Security* (CATARS), held in conjunction with the IEEE International Conference on Dependable Systems and Networks (DSN), 2008 and 2009. (one of two co-chairs)
- Delivered tutorial on “LLFI and the art of Fault Injection”
  - IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2017.
  - IEEE International Symposium on Software Reliability Engineering (ISSRE), 2019.
- Delivered tutorial on “Modern Web Applications’ Reliability Engineering”
  - IEEE International Symposium on Software Reliability Engineering (ISSRE), 2014.
  - IEEE International Conference on Quality, Reliability and Security (QRS), 2015.
  - IEEE International Symposium on Software Reliability Engineering (ISSRE), 2016.
- Selected Technical Program Committee member
  - o IEEE International Conference on Distributed Computing Systems (ICDCS), 2020.
  - o IEEE International Conference on Parallel Processing (ICPP), 2020.

# Curriculum Vitae: Karthik Pattabiraman

---

- o ACM International Conference on High-Performance and Distributed Computing (HPDC), 2019-2020.
- o IEEE/ACM International Conference on Automated Software Engineering (ASE), 2019 (Reliable, Rapid Response Reviewer).
- o IEEE International Conference on Dependable Systems and Networks (DSN), 2010-2012, 2015-2020.
- o IEEE International Symposium on Software Reliability Engineering (ISSRE), 2012, 2016-2019. 2020 (Program Board member)
- o IEEE/ACM International Conference on Design Automation (DAC), 2018.
- o International Conference on Computer Safety, Reliability and Security (SafeComp), 2016.
- o IEEE International Conference on Cloud Computing (Cloud), 2015, 2016.
- o IEEE International Online Testing Symposium (IOLTS), 2014, 2015.
- o IEEE International High Assurance Systems Symposium (HASE), 2010-2017.
- o IEEE Workshop on Silicon Errors in Logic, System Effects (SELSE), 2011-2014.
- o European Dependable Computing Conference (EDCC), 2017, 2018.
- o IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), 2012, 2015-2017.
- Expert Panel of Special issue of Journal of Software Systems (JSS), to select the best papers from ISSRE 2016.
- Reviewer for the following journals:
  - o IEEE Transactions on Dependable and Secure Computing (TDSC)
  - o IEEE Embedded Systems Letters
  - o ACM Journal on Emerging Technologies in Computing (JETC)
  - o IEEE Transactions on Computers
  - o IEEE Transactions on CAD (TCAD)
  - o IEEE Transactions on Cloud Computing
- Internal Service (UBC/ECE department)
  - o Space Committee (2011),
  - o Scholarship Committee (2012, 2013, 2014 – chair, 2019),
  - o Graduate Experience Committee (2013), Merit/PSA Committee (2013),
  - o Curriculum Committee (2015, 2016),
  - o Research Facilities Support Grant Committee (2017),
  - o External awards committee (2018),
  - o Faculty recruiting Committee (2018, 2019).

**Teaching Accomplishments:** I have taught 9 courses at UBC, three of which I created from scratch (CPEN400A, EECE571K, and EECE571P). Teaching evaluation scores from the last time I taught each course are provided out of 5.0, based on the answer to the question “Overall, the instructor was an effective teacher”. The ECE department average for undergraduate courses is around 3.7, while for graduate courses it is around 3.9.

## 1. Undergraduate courses (total of 5 courses taught over 6 years)

- a. Software Engineering (EECE 310) – Taught three times (Score: 4.0)
- b. Software Design (EECE 309/210) – Taught twice (Score: 4.0)
- c. Software Architecture (EECE 417) – Taught twice (Score: 3.8)
- d. Operating Systems (EECE 315) – Taught once (Score 4.4)
- e. Building Modern Web Applications (CPEN 400A) – Created, and Taught thrice (Score: 4.3)

## 2. Graduate Courses (total of 4 courses)

# Curriculum Vitae: Karthik Pattabiraman

---

- a. Design of Fault-tolerant Systems (EECE 513) – Taught three times (Score: 4.2)
- b. Program Analysis and Optimization (EECE 571P) – Created and Taught three times (Score: 4.4)
- c. Error Resilient Computing Systems (EECE 513) – Created and Taught twice (Score: 4.5)
- d. Security and Reliability of Internet of Things (EECE 571K/R) – Created and Taught twice (Score: 4.5)

## Research Accomplishments

1. **Good Enough Dependability:** This has been the mainstay of my research programme at UBC since I joined in 2010. Together with my students, I proposed the idea of trading off protection overhead for fault coverage in applications by systematically modeling error propagation, and reasoning about which errors are the most impactful. Prior to this approach, the default approach was “all-or-nothing” protection, which was used only in a few highly critical systems such as banking and healthcare, due to its high cost. In contrast, the good enough approach can be customized based on the application’s needs. Our main contribution has been to identify the important portions of an application for protection using both analytical modeling and empirical techniques, without requiring any support from the programmer. The good enough approach has since become the mainstream approach for dependability, and has been adopted by many companies as well. This line of work has received a best-paper nomination at DSN’18, and a distinguished paper award at EDCC’16.
2. **Empirical Studies of Reliability:** Along with my graduate students and colleagues at MSR and UBC, I pioneered the area of empirical studies of JavaScript-based web applications’ reliability. Our papers in this area showed that even mature, production websites exhibit significant numbers of JavaScript errors, and that the majority of these errors are caused by the interaction of the JavaScript code with the Document Object Model (DOM). The latter result flies in the face of conventional wisdom which rules that JavaScript errors occur due to the loosely typed nature of the language. This work has shaped many of the web application engineering tools we have since developed and released. We also performed empirical studies of cloud application reliability, which received an impact paper award at ISSRE’14.
3. **Fault Injectors:** My group at UBC has developed many fault injectors to evaluate the error resilience of programs. One of our main products in this space is LLFI, a fault injector based on the LLVM compiler that is highly configurable and allows easy mapping of the fault propagation results to the program’s code. LLFI supports a wide variety of failure modes in hardware and software, and is used by both researchers and industry. For example, Cisco Systems funded a project to adopt LLFI in their internal quality assurance process. It has also been used by IBM Research and Nvidia. Another fault injector developed by our group, GPU-Qin, to evaluate the error resilience of GPGPU applications, has been used by academic research groups, national labs, and industry research labs such as Nvidia and AMD.
4. **Tools for web application engineering:** We have developed a number of tools and techniques for web applications’ engineering based on the insights from our empirical studies. These have been published in top-tier international venues, and are being used by other researchers in the area. One such tool, Clematis, developed in collaboration with the SALT Lab at UBC, has won the SIGSOFT distinguished paper award at ICSE’14. We have also won best paper nominations/runner-up awards at ICST’12, and ICST’13.
5. **Internet of Things (IoT) Security:** In this line of work, we have developed a systematic framework for reasoning about the security of IoT devices such as smart meters, insulin pumps, drones etc. We have built a formal model for capturing the key behavioral properties of an IoT device, and then use the model to drive automated attack generation against the device. We have also built intrusion detection systems (IDSes) to efficiently detect attacks against IoT devices. This work has received a distinguished paper award at EDCC’15, and has received significant media coverage (see Press).
6. **Dependability of Machine Learning:** This is also a recent area I have been exploring with my students and colleagues at Nvidia and Los Alamos National Labs. We investigate the resilience of Machine Learning (ML) algorithms deployed on special purpose hardware platforms to soft errors. Our paper on the reliability of deep

# Curriculum Vitae: Karthik Pattabiraman

---

neural network (DNN) accelerators has been adopted by Nvidia research in their product. We have also built a fault injection tool for ML programs written using the TensorFlow framework, called TensorFI, in collaboration with Los Alamos National Labs. Recently, we built BinFI, a tool for optimizing fault injections into ML programs deployed in safety-critical systems such as self-driving cars.

**A more detailed research statement can be found on my homepage (<http://blogs.ubc.ca/karthik/>)**

## Selected Recent Talks (2010 to current)

- *Error Resilient Machine Learning for Safety-critical Applications*
  - IFIP WG 10.4 meeting on Machine learning dependability, June 2019.
  - Chinese University of Hong Kong, July 2019.
  - AMD Research, October 2019.
  - Workshop on Robust and Trustworthy Machine Learning, November 2019 (invited).
- *Resilience and Security in Cyber-Physical Systems: Self-Driving Cars and Smart Devices*,
  - Johns Hopkins University, March 2019.
  - University of Virginia, March 2019.
- *Error Resilience of Deep Neural Network Accelerators and Applications*, Trends in HPC Workshop, co-held with the HPDC PC meeting, Arlington, Virginia, March 2019.
- *Building Reliable Software for the Web, IoT and Beyond*,
  - Zhejiang University, China. May 2018.
  - Technische Universitat, Darmstadt, Germany, June 2018.
- *Modeling Hardware Error Propagation In Programs for Low Cost Dependability*, Northeastern University, Boston, 2018.
- *Resilience and Security in Cyber-Physical Systems: Self-Driving Cars and Smart Devices*, Microsoft Research (Redmond), 2017.
- *Tolerating Hardware Faults in Commodity Software: Problems, Solutions and a Roadmap*, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2017, Boston, US. *Opening Keynote speech*.
- *Why do Modern Web Applications Fail and What Can we Do About it?*,
  - CS department, Univ of Utah, 2016.
  - CS department, Univ of Massachusetts, Amherst, 2018.
- *Security and Reliability of the Internet of Things: A Smart Meter Case Study*, Microsoft Research, 2016.
- *Does Error Resilience matter in the age of Approximate Computing?* Invited Panelist at Workshop on Silicon Errors in Logic, Systems Effects (SELSE), 2016 on “Resilience and Probabilistic Computing”.
- *Error Resilient Systems and Approximate Computing: Conjoined Twins Separated at Birth?* Dagstuhl workshop on Approximate and Probabilistic Computing, November 2015 (invited).
- *Good Enough Dependability: A new paradigm for Dependable Systems Design*,
  - Purdue University, West Lafayette, Indiana, November 2017.
  - University of Illinois at Urbana-Champaign, October 2017.
  - Indian Institute of Technology (IIT), Madras, August 2017.
  - Universitat De Polytechnica, Valencia (UPV), Jan 2017.
  - University of Virginia (U. Va.), March 2016.
  - University of Illinois at Chicago (UIC), Feb 2016.
  - Rutgers, State University of New Jersey, October 2016. ECE Colloquium.
  - Microsoft Research, August 2015.
  - University of Maryland (College Park), February 2015.

# Curriculum Vitae: Karthik Pattabiraman

---

- o Technische University Darmstadt, November 2014.
- *Application-level Error Resilience: Tolerating Hardware Faults through Software Techniques*, Nvidia, 2015.
- *Good Enough Dependability: Providing Security and Reliability at Low Cost for Embedded Devices*, Qualcomm Research Silicon Valley, May 2015.
- *Helping Developers Understand, Analyze and Synthesize JavaScript Code in Web Applications*, Intel, May 2015.
- *Tolerating Silent Data Corruption (SDC) causing Hardware Faults through Software Techniques*,
  - o Electrical and Computer Engineering, Georgia Tech, Computer Engineering Seminar, June 2014.
  - o IBM T.J. Watson Research, New York, August 2014.
  - o AMD Research, Boston, August 2014.
- *Invited Panelist: “Towards 21<sup>st</sup> Century High Assurance Engineering”*, 15<sup>th</sup> IEEE High Assurance System Engineering (HASE), 2014, Miami, Florida. (one of four panelists)
- *How I learned to Stop Worrying and Love the DOM*,
  - o IFIP WG 2.4 Meeting, Victoria (April 2016)
  - o Microsoft Research, Redmond (August, 2014)
  - o Intel, Santa Clara (May 2013)
  - o Microsoft Research, India (August 2013)
  - o IBM Research, India (August 2013)
- *Why do Web Applications Fail and What can we do about it?*,
  - o Intel, Santa Clara, September 2012.
  - o CS department Colloquium, Queens University, August 2012.
  - o ECE department, University of Illinois at Urbana-Champaign (UIUC), May 2012.
  - o Microsoft Research, Redmond, June, 2011.
- *Good Enough Software Systems: Tolerating (most) Hardware Errors in Software*,
  - o CS department Colloquium, University of Pittsburgh, Feb 2012.
  - o CS department Colloquium, McGill University, Montreal, Quebec, March 2011.

## Press Coverage of Research

### 1. Drones and Rovers Security (December 2019)

- a. EurekaAlert ([https://www.eurekaalert.org/pub\\_releases/2019-11/uobc-urh112719.php](https://www.eurekaalert.org/pub_releases/2019-11/uobc-urh112719.php))
- b. TechXplore (<https://techxplore.com/news/2019-11-highlights-safeguard-drones-robotic-cars.html>)
- c. GlobalNews BC (<https://globalnews.ca/news/6235460/ubc-drone-hacking-research/>)
- d. Market Associates

(<https://themarketassociates.com/2019/12/03/the-need-to-secure-drones-and-automated-vehicles-against-cyber-attacks/>)

- e. HelpNet (<https://www.helpnetsecurity.com/2019/12/02/hacking-robotic-vehicles/>)

### 2. Smart meter security (June 2019)

- a. UBC (<https://news.ubc.ca/2019/06/06/ubc-researchers-find-ways-to-hackproof-smart-meters/>)
- b. TheStraight

(<https://www.straight.com/tech/1250511/smart-meters-your-home-are-target-hackers-says-ubc-researcher>)

- c. EurekaAlert ([https://www.eurekaalert.org/pub\\_releases/2019-06/uobc-urf060519.php](https://www.eurekaalert.org/pub_releases/2019-06/uobc-urf060519.php))
- d. E&T

(<https://eandt.theiet.org/content/articles/2019/06/system-developed-to-boost-smart-meter-resilience-to-cyber-attacks/>)

- e. ScienceDaily (<https://www.sciencedaily.com/releases/2019/06/190606101822.htm>)

- f. TechXplore (<https://techxplore.com/news/2019-06-ways-hackproof-smart-meters.html>)



# Curriculum Vitae: Karthik Pattabiraman

---

## Research Funding (Includes grants currently held and held in the past – amounts are specified in CAD<sup>1</sup>)

1. Building Error-Resilient Applications on Many-Core Platforms, NSERC Discovery Grant, 2010-2015, \$120,000
2. Microsoft Research, Unrestricted Gift, 2010, \$17,000.
3. Teaching and Learning Enhancement Fund, University of British Columbia, 2011 (co-PI with 2 others), \$39,000.  
NOTE: these grants are awarded on a competitive basis by UBC to enhance the undergraduate curriculum.
4. WATERS: A computational Infrastructure for Web Application Testing, Energy-Efficiency, Reliability and Security, Leaders Opportunity Funds (LOF), Canada Foundation for Innovation (CFI), 2011 (PI, 1 co-PI), \$280,000
5. Shared-memory Multiprocessor for Parallel Algorithms and Architectures, NSERC Research Tools and Infrastructure Grant (RTI), 2011. (Co-PI with six others). \$80,000.
6. Leveraging Dynamic Co-Processor Platforms for High-Performance Computing Applications, NSERC Engage Grant, 2011 (Sole PI). \$22,750. (Industry partner: Secodix, Vancouver).
7. Assessing the Error Reporting Capabilities of the Freescale QoRIQ Platform, NSERC Engage Grant, 2011 (Sole PI), 24,675. (Industry partner: Freescale).
8. Nokia Research, Unrestricted Gift, 2012. \$15,000.
9. Teaching and Learning Enhancement Fund, University of British Columbia, 2012 (co-PI, 1 PI) \$31,000.
10. Automatic Code Partitioning for XPU Acceleration, MITACS Accelerate Grant, 2012 (Sole PI), \$30,000.
11. Hardening Software to Detect Hardware Faults, NSERC Engage Grant, 2012 (Sole PI). \$25,000. (Industry partner: Cisco).
12. Systematic Software Analysis and Maintenance Techniques for Web 2.0 Applications, NSERC Strategic Project Grant, 2012-2015 (Co-PI with three others), \$480,000. My share: \$120,000
13. Unifying Static and Dynamic Analysis Techniques for Checking Non-Functional Properties, Lockheed Martin Research Grant, 2012-2013 (Co-PI with four others), \$180,000.
14. Secure and Trusted Network Terminals for Smart vehicular Networks, DIVA Strategic Network Grant, 2010-2015 (Multiple PIs, my portion is \$20,000 a year).
15. Software Robustness Assessment through Fault Injection, Research Grant, Cisco Systems, 2013-2015 (Sole PI). \$120,000.
16. Hardware Fault Injection for GPGPU Applications, NSERC Engage Grant, 2013 (Sole PI). \$25,000. (Industry partner: AMD).
17. Nvidia Equipment Donation, 2013. \$3000 market value.
18. Programming Techniques for QUBO compatible processors II, MITACS Accelerate Grant (Sole PI), 2013. \$30,000. (Industry partner: 1-Qbit, Vancouver).
19. Intel, Unrestricted Gift, 2012-2016 (PI, one co-PI), \$270,000. My share: \$135,000. NOTE: The funding was awarded each year on a competitive basis for four years in a row.
20. AMD, Unrestricted gift, 2014. \$8000.
21. Building Error-Resilient Applications on Next Generation Computing Platforms, NSERC Discovery Grant, 2015-2020, \$250,000 (Awarded an additional Discovery Accelerator (DAS) Supplement of \$120,000 for 2015-2018).
22. Low-cost Security for Internet of Things (IoT) Security, NSERC Engage Grant, 2015 (Sole PI), \$25,000. (Industry partner: Sierra Wireless).
23. Error Propagation Analysis for GPGPU Applications, NSERC Engage Grant, 2015 (Sole PI), \$23,000. (Industry partner: Nvidia).
24. Automatic Secure Code Migration for the Web of Things, Intel Research, 2016-2019 (PI, 1 Co-PI). \$210,000. My share: \$105,000.

---

<sup>1</sup> Unlike most US Government agencies, grant amounts from most Canadian Government agencies (e.g., NSERC) do not include overheads (i.e., indirect costs of research). Also, student fees are typically not included in the grant amounts, so grant sizes tend to be smaller as well.

# Curriculum Vitae: Karthik Pattabiraman

---

25. Error-Resilient Machine Learning Systems, NSERC Strategic Grant, 2017-2020 (Co-PI, 4 other co-PIs). \$720,000. My share: \$180,000
26. Designing Efficient and Resilient Deep Learning Accelerators using an AI Supercomputer, NSERC RTI Grant, 2018 (Co-PI, 4 other co-PIs). \$150,000. (Equipment grant)
27. Invariant-Driven Intrusion Detection System for Cyber-Physical Systems, NSERC Engage Grant, 2018. (Industry Partner: General Dynamics, Canada). \$25,000 (sole PI).
28. Automatic, Secure, Code Migration in the Heterogeneous world of the Internet of Things (IoT), NSERC CRD Grant, 2019. (PI, Co-PI: 2 Others) \$254,400. My share: \$90,000.
29. Huawei-UBC AI System-On-Chip (SoC) Program, \$71,400 (sole PI).
30. Huawei-UBC Software Engineering Research Program, \$72,800 (sole PI).

**Publications (by type, in chronological order): Names of students and post-docs I have advised are bolded.**

**As of Feb 2020, my h-index = 33. ([https://scholar.google.ca/citations?user=p\\_V9YWgAAAAJ&hl=en](https://scholar.google.ca/citations?user=p_V9YWgAAAAJ&hl=en))**

(a) *Journals: In my area, conference papers are often more prestigious and rigorous than journal papers. Also, many journal papers are expanded versions of the conference papers – these are indicated below.*

- [1] *Improving the accuracy of IR-level Fault Injection*, **Lucas Palazzi, Guanpeng Li, Bo Fang**, and Karthik Pattabiraman, IEEE Transactions on Dependable and Secure Computing (TDSC). Accept date: February 2020. (Expanded version of [18])
- [2] *Design-Level and Code-Level Security Analysis of IoT Devices*, **Farid Molazem Tabrizi** and Karthik Pattabiraman, ACM Transactions on Embedded Computing Systems (TECS). Accept date: January 2019. (Expanded version of [31]).
- [3] *Configurable Detection of SDC-causing Errors in Programs*, **Qining Lu, Guanpeng Li**, Karthik Pattabiraman, Meeta Gupta and Jude Rivers, ACM Transactions on Embedded Computing Systems (TECS), 2017. (Expanded version of [47]).
- [4] *A Study of Causes and Consequences of Client-Side JavaScript Bugs*, **Frolin Ocariza, Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, IEEE Transactions on Software Engineering (TSE), 2016. (Expanded version of [56])
- [5] *A Systematic Methodology for Evaluating the Error Resilience of GPGPU Applications*, **Bo Fang**, Karthik Pattabiraman, Matei Ripeanu, and Sudhanva Gurusurthy, IEEE Transactions on Parallel and Distributed Systems (TPDS). Accept date: December 2015. (Expanded version of [53]).
- [6] *Understanding JavaScript Event-Based Interactions with Clematis*, **Saba Alimadadi**, Ali Mesbah and Karthik Pattabiraman, ACM Transactions on Software Engineering and Methodology (TOSEM), 2016. (Expanded version of [51]).
- [7] *Error Detector Placement for Soft-Computing Applications*, **Anna Thomas** and Karthik Pattabiraman, ACM Transactions on Embedded Computing Systems (TECS), 2016. (expanded version of [57]).
- [8] *Automatic Fault Localization for Client-Side JavaScript*, **Frolin Ocariza, Guanpeng Li**, Karthik Pattabiraman and Ali Mesbah, Journal of Software Testing, Verification and Reliability (STVR), 2015. Accept Date: March 2015. (expanded version of [61]).
- [9] *Guided Mutation Testing for JavaScript Web Applications*, **Shabnam Mirshokraie**, Ali Mesbah and Karthik Pattabiraman, IEEE Transactions on Software Engineering (TSE), 41(5), 429-444 (2015) (expanded version of [58]).
- [10] *Characterizing the Impact of Intermittent Hardware Faults on Programs*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, IEEE Transactions on Reliability (TR), **64(1)**: 297-310, 2015 (expanded version of [65]).
- [11] *Modular Protections against Non-control Data Attacks*, Cole Schlesinger, Karthik Pattabiraman, Nikhil Swamy, David Walker, and Benjamin Zorn, Journal of Computer Security (JCS), **22(5)**: 699-742, 2014 (expanded version of [63]). **Invited as one of the best papers from CSF'11.**

# Curriculum Vitae: Karthik Pattabiraman

---

- [12] *SymPLFIED: Symbolic Program Level Fault Injection and Error Detection*, Karthik Pattabiraman, Nitin Nakka, Zbigniew Kalbarczyk and Ravishankar K Iyer, IEEE Transactions on Computers (TC), 62(11), 2013. (expanded version of [11]).
- [13] *Efficient Runtime Detection and Toleration of Asymmetric Races*, Paruj Ratanaworabhan, Martin Burtscher, Darko Kirovski, Benjamin Zorn, Rahul Nagpal, Karthik Pattabiraman, IEEE Transactions on Computers (TC), 61(4), 2012. (expanded version of [69]).
- [14] *Automated Derivation of Application-specific Error Detectors using Dynamic Analysis*, Karthik Pattabiraman, Giacinto Paolo Saggese, Daniel Chen, Zbigniew Kalbarczyk and Ravishankar Iyer, IEEE Transactions on Dependable and Secure Computing (TDSC), 8(5), 2011, (expanded version of [74]).
- [15] *Automated Derivation of Application-aware Error Detectors using Static Analysis*, Karthik Pattabiraman, Zbigniew Kalbarczyk and Ravishankar Iyer, IEEE Transactions on Dependable and Secure Computing (TDSC), 8(1), 2011. (expanded version of [72]).
- (b) *Conference Proceedings (Acceptance rates provided where known: see note above): Publications in tier-1 conferences are indicated with a '\*'. These are highly competitive venues with low acceptance rates.*
- [16] *\*Trace Sanitizer: Eliminating the Effects of Non-Determinism on Error Propagation Analysis*, Habib Saissi, Stefan Winter, Oliver Schwan, **Karthik Pattabiraman**, and Neeraj Suri, To appear in the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2020. (Acceptance Rate: 16.5%).
- [17] *Out of Control: Stealthy Attacks on Robotic Vehicles Protected by Control-Based Techniques*, **Pritam Dash, Mehdi Karimi**, and Karthik Pattabiraman, Proceedings of the Annual Computer Security Applications Conference (ACSAC), 2019. (Acceptance Rate: 22.6%).
- [18] *A tale of two injectors: An end-to-end comparison of IR-level and Assembly-level Fault Injection*, **Lucas Palazzi, Guanpeng Li, Bo Fang**, and Karthik Pattabiraman, Proceedings of the IEEE International Conference on Software Reliability Engineering (ISSRE), 2019. (Acceptance Rate: 31.5%).
- [19] *\*BinFI: An Efficient Fault Injector for Safety-Critical Machine Learning Systems*, **Zitao Chen, Guanpeng Li**, Karthik Pattabiraman, and Nathan DeBardeleben, Proceedings of the IEEE/ACM International Conference for High-Performance Computing, Storage and Networking (SC), 2019. (Acceptance Rate: 21%).
- [20] *BonVoision: Leveraging Spatial Data Smoothness for Recovery from Memory Soft Errors*, **Bo Fang**, Karthik Pattabiraman, Matei Ripeanu and Sriram Krishnamoorthy, Proceedings of the ACM International Conference on Supercomputing (ICS), 2019. (Acceptance Rate: 23.2%).
- [21] *Failure Prediction in the Internet of Things due to Memory Exhaustion*, **Mohammad Rafiuzzaman, Julien Gascon-Samson**, Karthik Pattabiraman, and Sathish Gopalakrishnan, Proceedings of the ACM/SIGAPP Symposium on Applied Computing (SAC), 2019. Dependable, Adaptive and Secure Distributed Systems Track. (Acceptance rate: 27.5%).
- [22] *DynPolAC: Dynamic Policy-based Access Control for IoT Systems*, **Mehdi Karimi, Ekta Aggarwal**, Karthik Pattabiraman, and Andre Ivanov, Proceedings of the IEEE International Conference on Pacific Rim Dependable Computing (PRDC), 2018. (Acceptance Rate: 45%).
- [23] *ThingsMigrate: Platform-Independent Migration of Stateful JavaScript IoT Applications*, Julien Gascon-Samson, Kumseok Jung, Shivanshu Goyal, Armin Rezalean-Asel, Karthik Pattabiraman, Proceedings of the European Conference on Object Oriented Programming (ECOOP), 2018. (Acceptance Rate: 39%).
- [24] *\*Modeling Soft Error Propagation in Programs*, **Guanpeng Li**, Karthik Pattabiraman, Siva Hari, Michael Sullivan and Timothy Tsai, Proceedings of the IFIP/IEEE International Conference on Dependable Systems and Networks (DSN), 2018 (Acceptance rate for regular papers: 25%). **Best Paper Award Nominee (1 of 3)**.
- [25] *\*Modeling Input-Dependent Error Propagation in Programs*, **Guanpeng Li**, and Karthik Pattabiraman, Proceedings of the IFIP/IEEE International Conference on Dependable Systems and Networks (DSN), 2018 (Acceptance rate for regular papers: 25%).
- [26] *\*Inferring Hierarchical Motifs from Execution Traces*, **Saba Alimadadi**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the ACM/IEEE International Conference on Software Engineering (ICSE), 2018 (Acceptance Rate: 21%).

## Curriculum Vitae: Karthik Pattabiraman

---

- [27] *\*Detecting Unknown Inconsistencies in Web Applications*, **Frolin Ocariza**, Karthik Pattabiraman, and Ali Mesbah, Proceedings of the ACM/IEEE International Conference on Automated Software Engineering (ASE), 2017. (Acceptance Rate: 21%)
- [28] *\*Understanding Error Propagation in Deep Learning Neural Network (DNN) Accelerators and Applications*, **Guanpeng Li**, Siva Hari, Michael Sullivan, Timothy Tsai, Karthik Pattabiraman, Steve Keckler, and Joel Emer, Proceedings of the International Conference for High-Performance Computing, Storage and Networking (SC), 2017. (Acceptance Rate: 19%)
- [29] *\*ARTINALI: Dynamic Invariant Detection for Cyber-Physical System Security*, **Maryam Raiyat Aliabadi, Amita Kamath, Julien Gascon-Samson**, and Karthik Pattabiraman, Proceedings of the ACM SIGSOFT Symposium on Foundations of Software Engineering (FSE), 2017. (Acceptance Rate: 24.5%)
- [30] *\*LetGo: A Lightweight Continuous Framework for HPC Applications Under Failures*, **Bo Fang**, Qiang Guan, Nathan Debardeleben, Karthik Pattabiraman, and Matei Ripeanu, Proceedings of the ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC), 2017. (Acceptance Rate: 19%)
- [31] *\*One Bit is (Not) Enough: An Empirical Study of the Impact of Single and Multiple Bit Flip Errors*, **Behrooz Sangchoolie**, Karthik Pattabiraman and Johan Karlsson, Proceedings of the IEEE International Conference on Dependable Systems and Networks (DSN), 2017. (Acceptance Rate: 23%)
- [32] *IPA: Error Propagation Analysis of Multi-Threaded Programs Using Likely Invariants*, **Abraham Chan**, Stefan Winter, Habib Saissi, Karthik Pattabiraman and Neeraj Suri, Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST), 2017. (Acceptance Rate: 27%)
- [33] *Formal Security Analysis of Smart Embedded Systems*, **Farid Molazem Tabrizi** and Karthik Pattabiraman, Proceedings of the Annual Computer Security Applications Conference (ACSAC), 2016. (Acceptance Rate: 23%).
- [34] *\*Understanding Error Propagation in GPGPU Applications*, **Guanpeng Li**, Karthik Pattabiraman, Chen-Yong Cher and Pradip Bose, Proceedings of the International Conference for High-Performance Computing, Storage and Networking (SC), 2016. (Acceptance Rate: 18%).
- [35] *Finding Resilience-Friendly Compiler Optimizations Using Meta-Heuristic Search Techniques*, **Nithya Narayanamurthy**, Karthik Pattabiraman and Matei Ripeanu, Proceedings of the European Dependable Computing Conference (EDCC), 2016. (Acceptance Rate: 41%). **Best Paper Award (1 of 3)**.
- [36] *FIDL: A Fault Injection Description Language for Compiler-Based Tools*, **Maryam Raiyat Ailabadi** and Karthik Pattabiraman, Proceedings of the 35<sup>th</sup> International Conference on Computer Safety, Reliability and Security (SafeComp), 2016. (Acceptance Rate: 35%)
- [37] *\*ePVF: An Enhanced Program Vulnerability Factor Methodology for Cross-Layer Resilience Analysis*, **Bo Fang, Qining Lu**, Karthik Pattabiraman, Matei Ripeanu and Sudhanva Gurumurthi, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2016. (Acceptance Rate: 21%).
- [38] *Atrina: Inferring Unit Oracles from GUI Test Cases*, **Shabnam Mirshokraie**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST), 2016. (Acceptance Rate: 27%)
- [39] *\*Understanding Asynchronous Interactions in Full-Stack JavaScript*, **Saba Alimadadi**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the IEEE/ACM International Conference on Software Engineering (ICSE), 2016. (Acceptance Rate: 19%)
- [40] *Experience Report: An Application-Specific Checkpointing Technique for Minimizing Checkpoint Corruption*, **Guanpeng Li**, Karthik Pattabiraman, Chen-Yong Cher and Pradip Bose, Proceedings of the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2015. (Acceptance Rate: 32%)
- [41] *\*Synthesizing Web Element Locators*, **Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the IEEE/ACM International Conference on Automated Software Engineering (ASE), 2015. (Acceptance Rate: 20.6%)
- [42] *Flexible Intrusion Detection Systems for Memory-Constrained Embedded Systems*, **Farid Tabrizi** and Karthik Pattabiraman, Proceedings of the 11<sup>th</sup> European Conference on Dependable Computing (EDCC), 2015. (Acceptance Rate: 46%). **Distinguished paper award – one of three from 54 submissions.**
- [43] *\*Hybrid Change-Impact Analysis for JavaScript Applications*, **Saba Alimadadi**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the European Conference on Object Oriented Programming (ECOOP), 2015. (Acceptance rate: 22.8%)

## Curriculum Vitae: Karthik Pattabiraman

---

- [44] *\*Fine Grained Characterization of Faults Causing Long Latency Crashes in Programs*, **Guanpeng Li, Qining Lu**, and Karthik Pattabiraman, Proceedings of the IEEE/IFIP International Conference on Dependable Systems (DSN), 2015. (Acceptance Rate: 22.5%)
- [45] *JSEFT: Automated JavaScript Unit Test Generation*, **Shabnam Mirshokraie**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST), 2015. (Acceptance Rate: 25%). **Invited as one of the best papers in the conference to the Journal on Software Testing and Verification (STVR).**
- [46] *\*Finding Inconsistencies in JavaScript MVC Applications*, **Frolin Ocariza**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the IEEE/ACM International Conference on Software Engineering (ICSE), 2015. (Acceptance Rate: 18.5%)
- [47] *Failure Analysis of Jobs in Compute Clouds: A Google Cluster Case Study*, **Xin Chen**, Charng-da Lu and *Karthik Pattabiraman*, Proceedings of the 25th IEEE International Symposium on Software Reliability Engineering (ISSRE), 2014. (Acceptance rate: 25%). **Chosen as one of the “highlights of ISSRE” – one of 26 papers chosen from over 1000 papers in the 30 year history of the ISSRE conference (in 2019).**
- [48] *\*DOMpletion: DOM-Aware JavaScript Code Completion*, **Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the ACM International Conference on Automated Software Engineering (ASE), 2014. (Acceptance Rate: 20%)
- [49] *\*SDCTune: A Model for Predicting the SDC Proneness of an Application for Configurable Protection*, **Qining Lu, Karthik Pattabiraman**, Meeta S. Gupta and Jude A. Rivers, International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES), 2014. (Acceptance Rate: 30%)
- [50] *\*Integrated Hardware-Software Diagnosis for Intermittent Hardware Faults*, **Majid Dadashi, Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2014. (Acceptance Rate: 30%)
- [51] *\*Quantifying the Accuracy of High-Level Fault Injection Techniques for Hardware Faults*, **Jiesheng Wei, Anna Thomas, Guanpeng Li**, and *Karthik Pattabiraman*, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2014. (Acceptance Rate: 30%)
- [52] *\*Vejovis: Suggesting Fixes for JavaScript Faults*, **Frolin Ocariza**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the IEEE/ACM International Conference on Software Engineering (ICSE), 2014, Hyderabad, India (Acceptance Rate: 20%).
- [53] *\*Understanding JavaScript Event-Based Interactions*, **Saba Alimadadi, Sheldon Sequira**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the IEEE/ACM International Conference on Software Engineering (ICSE), 2014, Hyderabad, India (Acceptance Rate: 20%). **ACM SIGSOFT Distinguished Paper Award (9 of nearly 500 submissions).**
- [54] *Mining Questions Asked by Web Developers*, **Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the 11<sup>th</sup> Working Conference on Mining Software Repositories (MSR), 2014. (Acceptance Rate: 34%)
- [55] *GPU-Qin: A Methodology for Evaluating the Error Resilience of GPGPU Applications*, **Bo Fang**, Karthik Pattabiraman, Matei Ripeanu and Sudhanva Gurusurthi, Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2014. (Acceptance Rate: 30%).
- [56] *Model-based Intrusion Detection for Smart Meters*, **Farid M. Tabrizi** and *Karthik Pattabiraman*, Proceedings of the IEEE International Symposium on High Assurance Systems Engineering (HASE), 2014. Miami, USA (Acceptance rate: 30%).
- [57] *\*Pythia: Generating Test Cases with Oracles for JavaScript Applications*, **Shabnam Mirshokraie**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the ACM/IEEE International Conference on Automated Software Engineering (ASE), New Ideas Track, 2013. October 2013 (Acceptance rate: 23%).
- [58] *An Empirical Study of Client-Side JavaScript Bugs*, **Frolin Ocariza, Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, Proceedings of the IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), 2013 (Acceptance rate: 28 %).
- [59] *\*Error Detector Placement for Soft Computation*, **Anna Thomas** and Karthik Pattabiraman, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2013. (Acceptance Rate: 20%).

# Curriculum Vitae: Karthik Pattabiraman

---

- [60] *Efficient JavaScript Mutation Testing*, **Shabnam Mirshokraie**, Ali Mesbah and Karthik Pattabiraman, Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST), 2013. (Acceptance Rate: 25 %). **Ranked as the second best paper (i.e., first runner up) at the conference of over 150 submissions.**
- [61] *Intermittent Hardware Errors Recovery: Modeling and Evaluation*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Proceedings of the International Conference on Quantitative Evaluation of Systems (QEST), 2012. (Acceptance rate: Unknown).
- [62] *\*BlockWatch: Leveraging Similarity in Parallel Programs for Error Detection*, **Jiesheng Wei** and Karthik Pattabiraman, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2012 (Acceptance rate: 17%).
- [63] *AutoFlox: An Automatic Fault Localizer For JavaScript*, **Frolin Ocariza**, Karthik Pattabiraman and Ali Mesbah, IEEE International Conference on Software Testing, Verification and Validation (ICST), 2012. (Acceptance rate: 27%). **Nominated for best paper award (one of six), among more than 150 submissions.**
- [64] *JavaScript Errors in the Wild: An Empirical Study*, **Frolin Ocariza**, Karthik Pattabiraman and Benjamin Zorn, Proceedings of IEEE International Symposium on Software Reliability Engineering (ISSRE), 2011 (Acceptance Rate: 25%).
- [65] *Modular Protections against Non-control Data Attacks*, **Cole Schlesinger**, Karthik Pattabiraman, Nikhil Swamy, David Walker and Benjamin Zorn, Proceedings of the IEEE Computer Security Foundations (CSF) Symposium, 2011 (Acceptance Rate: 26%). **Invited for a special issue by Journal of Computer Security (JCS).**
- [66] *\*Flicker: Saving DRAM Refresh-power through Critical Data Partitioning*, **Song Liu**, Karthik Pattabiraman, Thomas Moscibroda and Benjamin Zorn, Proceedings of the ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2011. (Acceptance Rate: 20%).
- [67] *Modeling the Propagation of Intermittent Hardware Faults in Programs*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Proceedings of the IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), 2010. (Acceptance rate: 41.5%).
- [68] *DoDOM: Leveraging DOM Invariants for Robustness Testing of Web 2.0 Applications*, **Karthik Pattabiraman** and Benjamin Zorn, Proceedings of the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2010. (Acceptance rate: 32 %).
- [69] *\*An End-to-end Approach for the Automatic Derivation of Application-aware Error Detectors*, Galen Lyle, Shelley Chen, **Karthik Pattabiraman**, Zbigniew Kalbarczyk and Ravishankar Iyer, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2009. (Acceptance rate: 25%).
- [70] *Discovering Application-level Insider attacks using Symbolic Execution*, Karthik Pattabiraman, Nithin Nakka, Zbigniew Kalbarczyk and **Ravishankar Iyer**, Proceedings of the IFIP International Information Security Conference (SEC), 2009. (Acceptance Rate: 22%).
- [71] *ToleRace: Detecting and Tolerating Asymmetric Races*, **Paruj Ratanaworabhan**, Martin Burtscher, Darko Kirovski, Benjamin Zorn, Karthik Pattabiraman and Rahul Nagpal, Proceedings of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP), 2009. (Acceptance Rate: 24%).
- [72] *\*SymPLFIED: Symbolic Program Level Fault-Injection and Error Detection Framework*, **Karthik Pattabiraman**, Nithin Nakka, Zbigniew Kalbarczyk and Ravishankar Iyer, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2008. (Acceptance Rate: 25%).
- [73] *\*Samurai: Protecting Critical Heap Data in Unsafe Languages*, Karthik Pattabiraman, Vinod Grover and **Benjamin Zorn**, Proceedings of ACM European Systems Conference (EuroSys), 2008. (Acceptance Rate: 18%).
- [74] *Automated Derivation of Application-Aware Error Detectors using Static Analysis*, Karthik Pattabiraman, Zbigniew Kalbarczyk and **Ravishankar Iyer**, Proceedings of IEEE International Online Test Symposium (IOLTS), 2007.
- [75] *\*Processor-level Selective Replication*, **Nithin Nakka**, Karthik Pattabiraman and Ravishankar Iyer, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2007. (Acceptance Rate: 25%).
- [76] *Dynamic Derivation of Application-Specific Error Detectors and their Hardware Implementation*, Karthik Pattabiraman, Giacinto Paulo Saggese, Daniel Chen, Zbigniew Kalbarczyk, and Ravishankar Iyer, Proceedings of the European Dependable Computing Conference (EDCC), 2006. (Acceptance Rate: 27%).

## Curriculum Vitae: Karthik Pattabiraman

---

- [77] *Application-Based Metrics for Strategic Placement of Detectors*, Karthik Pattabiraman, Zbigniew Kalbarczyk and Ravishankar Iyer, Proceedings of the IEEE Symposium on Pacific Rim Dependable Computing (PRDC), 2005. (Acceptance Rate: 36.5%).
- [78] *\*Modeling Coordinated Checkpointing for Large-Scale Supercomputers*, Long Wang, Karthik Pattabiraman, Larry Votta, Chris Vick, Alan Wood, Zbigniew Kalbarczyk and Ravishankar Iyer, Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2005. (Acceptance Rate: 25%).
- [79] *Formal Reasoning of Various Categories of Widely Exploited Security Vulnerabilities by Pointer Taintedness Semantics*, Shuo Chen, Karthik Pattabiraman, Zbigniew Kalbarczyk and Ravishankar Iyer, Proceedings of the IFIP International Information Security Conference (SEC), 2004. (Acceptance Rate: 22%).
- (c) *Selected Workshop Papers (Number of pages given, where applicable)*
- [80] *OneOS: IoT Platform based on POSIX and Actors*, **Kumseok Jung, Julien Gascon-Samson**, and Karthik Pattabiraman, 2<sup>nd</sup> Usenix workshop on Hot Topics in Edge Computing (HotEdge), 2019. (7 pages).
- [81] *Towards analytically evaluating the error resilience of GPU Programs*, **Abdul Rehman Anwer, Guanpeng Li, Karthik Pattabiraman, Siva Kumar Sastry Hari, Michael Sullivan and Timothy Tsai**, IEEE Workshop on Silicon Errors in Logic, System Effects (SELSE), 2019. (6 pages).
- [82] *ThingsJS: Towards a Distributed and Self-Adaptable Cloud Edge Middleware*, **Julien Gascon-Samson, Kumseok Jung**, and Karthik Pattabiraman, Poster presentation at the ACM/IEEE Symposium on Edge Computing (SEC), 2018. (2 pages)
- [83] *ThingsMigrate - Platform-Independent Live-Migration of JavaScript Processes*, **Kumseok Jung, Julien Gascon-Samson**, and Karthik Pattabiraman, Demo presentation at the ACM/IEEE Symposium on Edge Computing (SEC), 2018. (2 pages)
- [84] *CORGIDS: A Correlation-based Generic Intrusion Detection System*, Ekta Aggarwal, Mehdi Karimbuiki, Karthik Pattabiraman and Andre Ivanov, 5<sup>th</sup> ACM International Workshop on Cyber-Physical System Security (CPS-SPC), co-located with the ACM International Conference on Computer and Communications Security (CCS), 2018. (11 pages)
- [85] *TensorFI: A Configurable Fault Injector for TensorFlow Applications*, Guanpeng Li, Karthik Pattabiraman, and Nathan DeBardeleben, 8th IEEE International Workshop on Software Certification (WoSoCER), 2018, co-located with the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2018.
- [86] *SmartJS: Dynamic and Self-Adaptable Runtime Middleware for Next-Generation IoT Systems*, **Julien Gascon-Samson, Mohammad Rafiuzzaman** and Karthik Pattabiraman, Poster Paper, ACM SIGPLAN Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH) (2 pages).
- [87] *ThingsJS: Towards a Flexible and Self-Adaptable Middleware for Dynamic and Heterogeneous IoT Environments*, **Julien-Gascon Samson, Mohammad Rafiuzzaman**, and Karthik Pattabiraman, Workshop on Middleware and Applications for the Internet of Things (M4IoT), co-located with the ACM/IFIP/Usenix Middleware Conference, 2017 (6 pages).
- [88] *SDC is in the Eye of the Beholder: A Survey and Preliminary Study*, **Bo Fang**, Panruo Wu, Qiang Guan, Nathan Debardeleben, Laura Monroe, Sean Blanchard, Zhizong Chen, Karthik Pattabiraman, and Matei Ripeanu, International Workshop on Reliability and Security Data Analysis (RSDA), September 2016 (4 pages). Held in conjunction with the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2016.
- [89] *Intrusion Detection Systems for Embedded Systems*, **Farid Molazem Tabrizi** and Karthik Pattabiraman, Doctoral student Symposium Track of the ACM Middleware Conference (Middleware), 2015. (6 pages).
- [90] *A Fault Injection Description Language (FIDL) for Compiler-based Tools*, **Maryam Raiyat**, Karthik Pattabiraman and Nematollah Bidokhti, Industry Track of the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2015. (1 page – refereed based on abstract)
- [91] *LED: Tool for Synthesizing Web Element Locators*, **Kartik Bajaj**, Karthik Pattabiraman and Ali Mesbah, Tools Track of the IEEE/ACM International Conference on Automated Software Engineering (ASE), 2015. (3 pages)

## Curriculum Vitae: Karthik Pattabiraman

---

- [92] *LLFI: An Intermediate Code Level Fault Injector for Hardware Faults*, **Qining Lu, Mostafa Farahani, Jiasheng Wei, Anna Thomas** and Karthik Pattabiraman, Proceedings of IEEE International Conference on Quality, Reliability and Security (QRS), August 2015. Short paper. (6 pages)
- [93] *Failure Prediction of Jobs in Compute Clouds: A Google Cluster Case Study*, **Xin Chen**, Charng-da Lu and Karthik Pattabiraman, International Workshop on Reliability and Security Data Analysis (RSDA), 2014 (6 pages). Held in conjunction with the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2014. (6 pages)
- [94] *Soft-LLFI: A Comprehensive Framework for Software Fault Injection*, **Maryam Raiyat**, Karthik Pattabiraman and Nematollah Bidokhti, Industry Track of the IEEE International Symposium on Software Reliability Engineering (ISSRE), 2014.
- [95] *Evaluating the Error Resilience of Parallel Programs*, **Bo Fang**, Karthik Pattabiraman, Matei Ripeanu and Sudhanva Gurumurthi, Workshop on Fault Tolerance for High-Performance at Extreme Scale (FTXS), 2014 (6 pages). In conjunction with DSN 2014.
- [96] *Effect of Compiler Optimizations on the Error Resilience of Soft Computing Applications*, **Anna Thomas** and Karthik Pattabiraman, First Workshop on Algorithm and Application Error Resilience (AER), 2013, in conjunction with ICS 2013 (8 pages).
- [97] *Predicting Job Completion Times Using System Logs in Supercomputers*, **Xin Chen**, Charng-da Lu and Karthik Pattabiraman, IEEE Workshop on Reliable and Security Data Analysis (RSDA), 2013, in conjunction with DSN 2013 (6 pages).
- [98] *LLFI: An Intermediate Code Level Injector for Soft Computing Applications*, **Anna Thomas** and Karthik Pattabiraman, IEEE Workshop on Silicon Errors in Logic, System Effects (SELSE), 2013. (6 pages).
- [99] *SCRIBE: A Hardware Infrastructure Enabling Fine-Grained Software Error Diagnosis*, **Majid Dadashi, Layali Rashid**, Karthik Pattabiraman, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2013. (6 pages).
- [100] *Towards Building Error Resilient GPGPU Applications*, **Bo Fang, Jiasheng Wei**, Karthik Pattabiraman, Matei Ripeanu, 3rd IEEE Workshop on Resilient Architecture (WRA) in conjunction with MICRO 2012. (6 pages).
- [101] *A Model for Security Analysis of Smart Meters*, **Farid M. Tabrizi** and Karthik Pattabiraman, 6<sup>th</sup> Workshop on Recent Advances in Intrusion Tolerance and Resilience (WRAITS), 2012, held with DSN 2012. (6 pages).
- [102] *DIEBA: Diagnosing Intermittent Errors by Backtracing Application Failures*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2012. (6 pages).
- [103] *BlockWatch: Leveraging Similarity in Parallel Programs for Error Detection*, **Jiasheng Wei** and Karthik Pattabiraman, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2012. (6 pages).
- [104] *Comparing the Effects of Intermittent and Transient Hardware Faults on Programs*, **Jiasheng Wei, Layali Rashid**, Karthik Pattabiraman, Sathish Gopalakrishnan, Workshop on Dependable and Secure Nano-computing (WDSN), 2011, in conjunction with DSN 2011 (6 pages).
- [105] *Towards Understanding the Effects of Intermittent Hardware Faults on Programs*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Workshop on Dependable and Secure Nanocomputing (WDSN), 2010, in conjunction with DSN 2010 (6 pages).
- [106] *Formal Diagnosis of Hardware Transient Errors in Programs*, **Layali Rashid**, Karthik Pattabiraman and Sathish Gopalakrishnan, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2010 (6 pages).
- [107] *Hardware Implementation of Information Flow Signatures Derived via Program Analysis*, Paul Dabrowski, William Healey, Karthik Pattabiraman, Shelley Chen, Zbigniew Kalbarczyk, Ravishankar Iyer, Workshop on Dependable and Secure Nano-computing (WDSN), 2008, In conjunction with DSN 2008 (6 pages).
- [108] *Critical Variable Recomputation for Transient Error Detection*, Karthik Pattabiraman, Zbigniew Kalbarczyk and Ravishankar K. Iyer, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2007. (6 pages).
- [109] *FPGA Hardware Implementation of Statically Derived Application-aware Error Detectors*, Peter Klemperer, Shelley Chen, Karthik Pattabiraman, Zbigniew Kalbarczyk and Ravishankar Iyer, Workshop on Dependable and Secure Nano-computing (WDSN), In conjunction with DSN 2007. (6 pages).
- [110] *Tolerance: Tolerating and Detecting Asymmetric Races (Position Paper)*, Rahul Nagpal, Karthik Pattabiraman, Darko Kirovski, Benjamin Zorn, Workshop on Software Tools for Multi-core Systems (STMCS), 2007 (4 pages), in conjunction with PPOPP 2007.



# Curriculum Vitae: Karthik Pattabiraman

---

[111] *Processor-level Selective Replication*, Nithin Nakka, Karthik Pattabiraman, Zbigniew Kalbarczyk, Ravishankar Iyer, Workshop on Silicon Errors in Logic, System Effects (SELSE), 2006. (6 pages).

[112] *Automated Derivation and Hardware Implementation of Application-Specific Error Detectors*, Karthik Pattabiraman, Giacinto Paulo Saggese, Daniel Chen, Zbigniew Kalbarczyk and Ravishankar Iyer, Workshop on Reliability Issues in High-Performance Computing (HPCRI), 2005, in conjunction with HPCA'05 (6 pages).

(d) *Invited contributions (non-refereed or lightly refereed)*

[113] Karthik Pattabiraman and **Guanpeng Li**, *Fault Injection at the Instruction Set Architecture (ISA) Level*, To appear as a chapter in the “*Cross Layer Reliability of Computer Systems*”, Editors: Giorgio De Natale, Dimitros Gizopoulos, Stefano De Carlo, Alberto Benso, and Ramon Canal, The Institution of Engineering and Technology (IET). Publication date early 2020 (expected).

[114] *GPUS: Combining high-performance with high-reliability*, L. Bautista Gomez, F. Cappello, L. Carro, N. DeBardeleben, **B. Fang**, S. Gurumurthi, K. Pattabiraman, P. Rech, M. Sonza Reorda, Embedded tutorial paper Proceedings of the International Symposium on Design Automation and Test in Europe (DATE), 2014.

[115] *Towards Application-aware Security and Reliability*, Ravishankar Iyer, Zbigniew Kalbarczyk, Karthik Pattabiraman, Wen-Mei Hwu, William Healey, Peter Klemperer and Reza Farivar, IEEE Security and Privacy (S&P) Magazine, Jan 2007. pages: 57- 62.

(e) *Patents*

[116] *Critical Memory*, with Benjamin Zorn and Vinod Grover, Microsoft Corporation, 2011.

[117] *Critical Memory using Replication*, with Benjamin Zorn, Vinod Grover, Microsoft Corporation, 2011.

[118] *Providing Hardware Resources having Different Reliabilities for Use by an Application*, with Benjamin Zorn, Thomas Moscibroda and Song Liu, Microsoft Corporation, 2016.

[119] *Leveraging On-Chip Variability*, Benjamin Zorn, Darko Kirovski, Ray Bittner, and Karthik Pattabiraman, Microsoft Corporation, 2016.

(f) *Software Artifacts Released (Most of these are released under a BSD/MIT/Illinois open source license)*

[120] *LLFI*: A fault injector based on the LLVM compiler. Available:

<https://github.com/DependableSystemsLab/LLFI>

[121] *GPU-Qin*: A fault injector for injecting faults into GPGPUs. Available:

<https://github.com/DependableSystemsLab/GPU-Injector>

[122] *PINFI*: A fault-injector based on Intel's PIN tool for injecting faults into program binaries. Available:

<https://github.com/DependableSystemsLab/PINFI>

[123] *AutoFlox*: Localize faults in JavaScript-based Web Applications. Available:

<http://ece.ubc.ca/~frolino/projects/autoflox/>

[124] *VejoVis*: Suggest fixes for faults in JavaScript-based Web Applications. Available:

<http://ece.ubc.ca/~frolino/projects/vejoVis/>

[125] *Clematis*: Enable understanding of JavaScript-based Web Applications. Available:

<http://salt.ece.ubc.ca/software/clematis/>

[126] *Mutandis*: Perform mutation testing in JavaScript Web Applications. Available:

<https://github.com/saltlab/mutandis>

[127] *Pythia*: Generate Oracles for test cases in JavaScript Web Applications. Available:

<https://github.com/saltlab/Pythia>

# Curriculum Vitae: Karthik Pattabiraman

---

- [128] *DOMpletion*: Enable automatic code completion in JavaScript Web Applications. Available:  
<https://github.com/saltlab/dompletion>
- [129] *CrashFinder*: Find long-latency causing crash locations in Programs through Static Analysis. Available:  
<https://github.com/DependableSystemsLab/CrashFinder>
- [130] *Aurebesh*: Find inconsistencies in MVC frameworks used in web applications. Available:  
<http://ece.ubc.ca/~frolino/projects/aurebesh/polyglot/index.html>
- [131] *LED*: Synthesize JavaScript code for DOM interactions in web applications. Available:  
<https://github.com/saltlab/led>
- [132] *ToChal*: Change Impact Analysis for JavaScript-based Web Applications. Available:  
<https://github.com/saltlab/tochal>
- [133] *JSEFT*: Automated Test and Oracle Generation for JavaScript-based Web applications. Available:  
<https://github.com/saltlab/JSeft>
- [134] *ReCov*: Fine-grained checkpointing without checkpoint corruption, Available:  
<https://github.com/DependableSystemsLab/ReCov>
- [135] *LLFI-GPU*: Fault injection tool for GPUs at the LLVM IR level. Available:  
<https://github.com/DependableSystemsLab/LLFI-GPU>
- [136] *ARTINALI*: Invariant detection for cyber-physical systems. Available:  
<https://github.com/DependableSystemsLab/Artinali>
- [137] *Trident/vTrident*: Analytical model for SDC rate prediction of programs. Available:  
<https://github.com/DependableSystemsLab/Trident>
- [138] *DNNFI*: Fault injector for deep learning neural network Accelerator platforms. Available:  
<https://github.com/DependableSystemsLab/DNNFI>
- [139] *LLFI-IPA*: Invariant-based Error Propagation Analysis Framework. Available:  
<https://github.com/DependableSystemsLab/LLFI-IPA>
- [140] *Sahand*: Understanding JavaScript Client-Server Asynchronous Interactions. Available:  
<https://github.com/saltlab/sahand>
- [141] *Holocron*: Inconsistency Detection Tool for JavaScript MVC applications. Available:  
<https://github.com/DependableSystemsLab/Holocron>
- [142] *Sabalan*: Extracting Motifs from Web Applications' Dynamic Traces. Available:  
<https://github.com/saltlab/sabalan>
- [143] *LetGo*: Roll-forward error recovery for parallel programs' crashes. Available:  
<https://github.com/DependableSystemsLab/LetGo>
- [144] *ThingsJS*: Adaptive runtime middleware for JavaScript on IoT devices. Available:  
<https://github.com/DependableSystemsLab/ThingsJS>
- [145] *DynPolAC*: Dynamic Policy Based Access Control for IoT. Available:  
<https://github.com/DependableSystemsLab/DynPolAC>
- [146] *TensorFI*: Fault Injector for TensorFlow Applications. Available:  
<https://github.com/DependableSystemsLab/TensorFI>
- [147] *OneOS*: Overlay Network Operating Systems  
<https://github.com/DependableSystemsLab/OneOS>
- [148] *BinFI*: Binary Fault Injector TensorFlow Applications  
<https://github.com/DependableSystemsLab/TensorFI-BinaryFI>
- [149] ACSAC'19: Stealthy attacks against Robotic Vehicles found in our ACSAC'19 paper

## Curriculum Vitae: Karthik Pattabiraman

---

- [150] <https://github.com/DependableSystemsLab/stealthy-attacks>  
TriInjectors: Results from our ISSRE'19 paper on fault injectors' comparison
- [151] <https://github.com/DependableSystemsLab/ISSRE19>  
PC Meeting Web Dashboard: For running physical PC meetings - used in DSN'19 , DSN'20 and the HPDC'19 PC meetings  
<https://github.com/DependableSystemsLab/PCMeetingDashboard>