

Michael Hicks

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/3717078/publications.pdf>

Version: 2024-02-01

107
papers

3,636
citations

394286

19
h-index

395590

33
g-index

110
all docs

110
docs citations

110
times ranked

1269
citing authors

#	ARTICLE	IF	CITATIONS
1	C to checked C by 3c. , 2022, 6, 1-29.		8
2	BullFrog: Online Schema Evolution via Lazy Evaluation. , 2021, , .		1
3	A verified optimizer for Quantum circuits. , 2021, 5, 1-29.		35
4	A language for probabilistically oblivious computation. , 2020, 4, 1-31.		7
5	Verifying replicated data types with typeclass refinements in Liquid Haskell. , 2020, 4, 1-30.		8
6	MVEDSUA. , 2019, , .		16
7	LWeb: information flow security for multi-tier web applications. , 2019, 3, 1-30.		17
8	Wys [*] : A DSL for Verified Secure Multi-party Computations. Lecture Notes in Computer Science, 2019, , 99-122.	1.0	10
9	Checked C: Making C Safe by Extension. , 2018, , .		24
10	What's the Over/Under? Probabilistic Bounds on Information Leakage. Lecture Notes in Computer Science, 2018, , 3-27.	1.0	6
11	Evaluating Design Tradeoffs in Numeric Static Analysis for Java. Lecture Notes in Computer Science, 2018, , 653-682.	1.0	9
12	Languages for Oblivious Computation. , 2017, , .		0
13	Quantifying Vulnerability of Secret Generation Using Hyper-Distributions. Lecture Notes in Computer Science, 2017, , 26-48.	1.0	0
14	Strider: type-aware heap traversal for C. Software - Practice and Experience, 2016, 46, 767-788.	2.5	5
15	Evolving NoSQL Databases without Downtime. , 2016, , .		22
16	Safe and Flexible Controller Upgrades for SDNs. , 2016, , .		9
17	Tedsuto: A General Framework for Testing Dynamic Software Updates. , 2016, , .		13
18	GhostRider. , 2015, , .		61

#	ARTICLE	IF	CITATIONS
19	GhostRider. ACM SIGPLAN Notices, 2015, 50, 87-101.	0.2	19
20	GhostRider. Computer Architecture News, 2015, 43, 87-101.	2.5	7
21	Incremental computation with names. , 2015, , .		22
22	Incremental computation with names. ACM SIGPLAN Notices, 2015, 50, 748-766.	0.2	5
23	Authenticated data structures, generically. , 2014, , .		28
24	Rubah. , 2014, , .		24
25	Quantifying Information Flow for Dynamic Secrets. , 2014, , .		34
26	Automating Efficient RAM-Model Secure Computation. , 2014, , .		70
27	Wysteria: A Programming Language for Generic, Mixed-Mode Multiparty Computations. , 2014, , .		70
28	Kitsune. ACM Transactions on Programming Languages and Systems, 2014, 36, 1-38.	1.7	22
29	Authenticated data structures, generically. ACM SIGPLAN Notices, 2014, 49, 411-423.	0.2	24
30	Adapton. , 2014, , .		39
31	Rubah. ACM SIGPLAN Notices, 2014, 49, 103-119.	0.2	2
32	Knowledge inference for optimizing secure multi-party computation. , 2013, , .		10
33	Dynamic enforcement of knowledge-based security policies using probabilistic abstract interpretation. Journal of Computer Security, 2013, 21, 463-532.	0.5	19
34	Memory Trace Oblivious Program Execution. , 2013, , .		37
35	Expositor: Scriptable time-travel debugging with first-class traces. , 2013, , .		6
36	Kitsune. , 2012, , .		45

#	ARTICLE	IF	CITATIONS
37	Automating object transformations for dynamic software updating. , 2012, , .		14
38	Kitsune. ACM SIGPLAN Notices, 2012, 47, 249-264.	0.2	8
39	Towards standardized benchmarks for Dynamic Software Updating systems. , 2012, , .		6
40	A study of dynamic software update quiescence for multithreaded programs. , 2012, , .		10
41	Evaluating Dynamic Software Update Safety Using Systematic Testing. IEEE Transactions on Software Engineering, 2012, 38, 1340-1354.	4.3	26
42	Specifying and Verifying the Correctness of Dynamic Software Updates. Lecture Notes in Computer Science, 2012, , 278-293.	1.0	45
43	Dynamic Enforcement of Knowledge-Based Security Policies. , 2011, , .		33
44	State transfer for clear and efficient runtime updates. , 2011, , .		28
45	LOCKSMITH. ACM Transactions on Programming Languages and Systems, 2011, 33, 1-55.	1.7	83
46	Dynamic inference of static types for ruby. ACM SIGPLAN Notices, 2011, 46, 459-472.	0.2	15
47	Lightweight monadic programming in ML. , 2011, , .		26
48	Directed Symbolic Execution. Lecture Notes in Computer Science, 2011, , 95-111.	1.0	95
49	Dynamically checking ownership policies in concurrent c/c++ programs. ACM SIGPLAN Notices, 2010, 45, 457-470.	0.2	3
50	Serializing C intermediate representations for efficient and portable parsing. Software - Practice and Experience, 2010, 40, n/a-n/a.	2.5	1
51	Verified enforcement of stateful information release policies. ACM SIGPLAN Notices, 2009, 43, 21-31.	0.2	3
52	Passive aggressive measurement with MGRP. , 2009, , .		10
53	The ruby intermediate language. , 2009, , .		15
54	Dynamic software updates. , 2009, , .		107

#	ARTICLE	IF	CITATIONS
55	Static type inference for Ruby. , 2009, , .		106
56	Safe and timely updates to multi-threaded programs. , 2009, , .		67
57	Passive aggressive measurement with MGRP. Computer Communication Review, 2009, 39, 279-290.	1.5	10
58	A theory of typed coercions and its applications. , 2009, , .		18
59	Directing JavaScript with arrows. , 2009, , .		7
60	Cross-tier, label-based security enforcement for web applications. , 2009, , .		37
61	Safe and timely updates to multi-threaded programs. ACM SIGPLAN Notices, 2009, 44, 13-24.	0.2	21
62	The ruby intermediate language. ACM SIGPLAN Notices, 2009, 44, 89-98.	0.2	1
63	Dynamic software updates. ACM SIGPLAN Notices, 2009, 44, 1-12.	0.2	29
64	Tests to the left of me, types to the right. , 2009, , .		0
65	A theory of typed coercions and its applications. ACM SIGPLAN Notices, 2009, 44, 329-340.	0.2	0
66	Modular Information Hiding and Type-Safe Linking for C. IEEE Transactions on Software Engineering, 2008, 34, 357-376.	4.3	2
67	Fable: A Language for Enforcing User-defined Security Policies. Sp'97, 2008, , .	0.0	73
68	Contextual effects for version-consistent dynamic software updating and safe concurrent programming. , 2008, , .		45
69	Verified enforcement of stateful information release policies. , 2008, , .		9
70	Dynamic rebinding for marshalling and update, via redex-time and destruct-time reduction. Journal of Functional Programming, 2008, 18, .	0.5	3
71	Formalizing Soundness of Contextual Effects. Lecture Notes in Computer Science, 2008, , 262-277.	1.0	1
72	Implicit Flows: Canâ€™t Live with â€™Em, Canâ€™t Live without â€™Em. Lecture Notes in Computer Science, 2008, 56-70.	1.0	54

#	ARTICLE	IF	CITATIONS
73	Contextual effects for version-consistent dynamic software updating and safe concurrent programming. ACM SIGPLAN Notices, 2008, 43, 37-49.	0.2	19
74	Path projection for user-centered static analysis tools. , 2008, , .		25
75	<i>Mutatis Mutandis</i>. ACM Transactions on Programming Languages and Systems, 2007, 29, 22.	1.7	65
76	Automated detection of persistent kernel control-flow attacks. , 2007, , .		189
77	Verified Enforcement of Security Policies for Cross-Domain Information Flows. , 2007, , .		6
78	Practical dynamic software updating for C. , 2006, , .		116
79	Safe manual memory management in Cyclone. Science of Computer Programming, 2006, 62, 122-144.	1.5	35
80	Trusted declassification:. , 2006, , .		23
81	LOCKSMITH. , 2006, , .		158
82	LOCKSMITH. ACM SIGPLAN Notices, 2006, 41, 320-331.	0.2	38
83	Practical dynamic software updating for C. ACM SIGPLAN Notices, 2006, 41, 72-83.	0.2	57
84	Merging Network Measurement with Data Transport. Lecture Notes in Computer Science, 2005, , 368-371.	1.0	3
85	Understanding source code evolution using abstract syntax tree matching. , 2005, , .		87
86	Dynamic software updating. ACM Transactions on Programming Languages and Systems, 2005, 27, 1049-1096.	1.7	136
87	Understanding source code evolution using abstract syntax tree matching. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2005, 30, 1-5.	0.5	82
88	Tagged Sets: A Secure and Transparent Coordination Medium. Lecture Notes in Computer Science, 2005, , 252-267.	1.0	5
89	Mutatis mutandis. , 2005, , .		31
90	Mutatis mutandis. ACM SIGPLAN Notices, 2005, 40, 183-194.	0.2	5

#	ARTICLE	IF	CITATIONS
91	Experience with safe manual memory-management in cyclone. , 2004, , .		57
92	Dynamic rebinding for marshalling and update, with destruct-time ?. , 2003, , .		21
93	Region-based memory management in cyclone. , 2002, , .		263
94	Evolution in Action: Using Active Networking to Evolve Network Support for Mobility. Lecture Notes in Computer Science, 2002, , 146-161.	1.0	6
95	Dynamic software updating. ACM SIGPLAN Notices, 2001, 36, 13-23.	0.2	9
96	Dynamic software updating. , 2001, , .		112
97	Safe and Flexible Dynamic Linking of Native Code. Lecture Notes in Computer Science, 2001, , 147-176.	1.0	12
98	Compiling PLAN to SNAP. Lecture Notes in Computer Science, 2001, , 134-151.	1.0	7
99	Active Networking Means Evolution (or Enhanced Extensibility Required). Lecture Notes in Computer Science, 2000, , 16-32.	1.0	10
100	Transparent communication for distributed objects in Java. , 1999, , .		7
101	Specifying the PLAN Network Programming Language. Electronic Notes in Theoretical Computer Science, 1999, 26, 87-104.	0.9	7
102	PLAN. ACM SIGPLAN Notices, 1999, 34, 86-93.	0.2	21
103	A Secure Plan. Lecture Notes in Computer Science, 1999, , 307-314.	1.0	31
104	A study of large object spaces. ACM SIGPLAN Notices, 1999, 34, 138-145.	0.2	3
105	PLAN. , 1998, , .		185
106	A study of large object spaces. , 1998, , .		17
107	Polymonadic Programming. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 153, 79-99.	0.8	11