Christian Colombo

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/3757505/publications.pdf

Version: 2024-02-01

47 papers	565 citations	840776 11 h-index	794594 19 g-index
56 all docs	56 docs citations	56 times ranked	263 citing authors

#	Article	IF	Citations
1	LARVA --- Safer Monitoring of Real-Time Java Programs (Tool Paper). , 2009, , .		99
2	A survey of challenges for runtime verification from advanced application domains (beyond) Tj ETQq0 0 0 rgBT /Overlock 10 Tf 50 702 T	0.8	56
3	Dynamic Event-Based Runtime Monitoring of Real-Time and Contextual Properties. Lecture Notes in Computer Science, 2009, , 135-149.	1.3	56
4	First international Competition on Runtime Verification: rules, benchmarks, tools, and final results of CRV 2014. International Journal on Software Tools for Technology Transfer, 2019, 21, 31-70.	1.9	48
5	Organising LTL monitors over distributed systems with a global clock. Formal Methods in System Design, 2016, 49, 109-158.	0.8	34
6	Safer asynchronous runtime monitoring using compensations. Formal Methods in System Design, 2012, 41, 269-294.	0.8	20
7	polyLarva: Runtime Verification with Configurable Resource-Aware Monitoring Boundaries. Lecture Notes in Computer Science, 2012, , 218-232.	1.3	18
8	Recovery within long-running transactions. ACM Computing Surveys, 2013, 45, 1-35.	23.0	17
9	LarvaStat: Monitoring of Statistical Properties. Lecture Notes in Computer Science, 2010, , 480-484.	1.3	13
10	Lessons learnt from using DSLs for automated software testing. , 2015, , .		12
11	Towards Incremental Mutation Testing. Electronic Notes in Theoretical Computer Science, 2013, 294, 2-11.	0.9	11
12	Fast-Forward Runtime Monitoring â€" An Industrial Case Study. Lecture Notes in Computer Science, 2013, , 214-228.	1.3	11
13	Contracts over Smart Contracts: Recovering from Violations Dynamically. Lecture Notes in Computer Science, 2018, , 300-315.	1.3	10
14	Industrial Experiences with Runtime Verification of Financial Transaction Systems: Lessons Learnt and Standing Challenges. Lecture Notes in Computer Science, 2018, , 211-232.	1.3	10
15	Verifying Web Applications: From Business Level Specifications to Automated Model-Based Testing. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 141, 14-28.	0.8	8
16	A Controlled Natural Language for Tax Fraud Detection. Lecture Notes in Computer Science, 2016, , 1-12.	1.3	6
17	A Model-Based Approach to Combining Static and Dynamic Verification Techniques. Lecture Notes in Computer Science, 2016, , 416-430.	1.3	6
18	Reducing the Forensic Footprint with Android Accessibility Attacks. Lecture Notes in Computer Science, 2020, , 22-38.	1.3	6

#	Article	IF	Citations
19	A Technique for Automata-based Verification with Residual Reasoning. , 2020, , .		6
20	CLARVA: Model-based Residual Verification of Java Programs. , 2020, , .		6
21	A Controlled Natural Language for Business Intelligence Monitoring. Lecture Notes in Computer Science, 2015, , 300-306.	1.3	5
22	An Automata-Based Approach to Evolving Privacy Policies for Social Networks. Lecture Notes in Computer Science, 2016, , 285-301.	1.3	4
23	Comprehensive Monitor-Oriented Compensation Programming. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 147, 47-61.	0.8	4
24	Control-Flow Residual Analysis for Symbolic Automata. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 254, 29-43.	0.8	4
25	Using gherkin to extract tests and monitors for safer medical device interaction design. , 2016, , .		3
26	Model-Based Static and Runtime Verification for Ethereum Smart Contracts. Communications in Computer and Information Science, 2021, , 323-348.	0.5	3
27	RV-TEE: secure cryptographic protocol execution based on runtime verification. Journal of Computer Virology and Hacking Techniques, 2021, 17, 229-248.	2.2	3
28	Investigating Instrumentation Techniques for ESB Runtime Verification. Lecture Notes in Computer Science, 2015, , 99-107.	1.3	3
29	Compliance Checking in the Open Payments Ecosystem. Lecture Notes in Computer Science, 2016, , 337-343.	1.3	3
30	Towards a Comprehensive Solution for Secure Cryptographic Protocol Execution based on Runtime Verification. , 2020, , .		3
31	Real-Time Triggering of Android Memory Dumps for Stealthy Attack Investigation. Lecture Notes in Computer Science, 2021, , 20-36.	1.3	2
32	Runtime Verification for Stream Processing Applications. Lecture Notes in Computer Science, 2016, , 400-406.	1.3	2
33	Simplifying Contract-Violating Traces. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 94, 11-20.	0.8	2
34	Device-Centric Monitoring for Mobile Device Management. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 205, 31-44.	0.8	2
35	Considering Academia-Industry Projects Meta-characteristics in Runtime Verification Design. Lecture Notes in Computer Science, 2018, , 32-41.	1.3	2
36	Responding to Targeted Stealthy Attacks on Android Using Timely-Captured Memory Dumps. IEEE Access, 2022, 10, 35172-35218.	4.2	2

#	Article	IF	Citations
37	Runtime verification for trustworthy secure shell deployment. , 2021, , .		1
38	SMock â€” A Test Platform for Monitoring Tools. Lecture Notes in Computer Science, 2013, , 352-357.	1.3	1
39	Extensible Technology-Agnostic Runtime Verification. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 108, 1-15.	0.8	1
40	Exploring the Link Between Test Suite Quality and Automatic Specification Inference. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 254, 44-56.	0.8	1
41	COST Action IC1402 Runtime Verification Beyond Monitoring. Lecture Notes in Computer Science, 2018, , 18-26.	1.3	1
42	Runtime Verification of Contracts with Themulus. Lecture Notes in Computer Science, 2020, , 231-246.	1.3	1
43	Themulus: A Timed Contract-calculus. , 2020, , .		1
44	Responding to Living-Off-the-Land Tactics using Just-In-Time Memory Forensics (JIT-MF) for Android. , 2021, , .		1
45	Using control flow analysis to improve the effectiveness of incremental mutation testing. , 2015, , .		0
46	Responding to Living-Off-the-Land Tactics using Just-In-Time Memory Forensics (JIT-MF) for Android. , 2021, , .		0
47	Runtime Verification: Passing on the Baton. Lecture Notes in Computer Science, 2021, , 89-107.	1.3	0