

Hee Beng Kuan Tan

List of Publications by Year in Descending Order

Source: <https://exaly.com/author-pdf/11363758/hee-beng-kuan-tan-publications-by-year.pdf>

Version: 2024-04-27

This document has been generated based on the publications and citations recorded by exaly.com. For the latest version of this publication list, visit the link given above.

The third column is the impact factor (IF) of the journal, and the fourth column is the number of citations of the article.

44
papers

521
citations

11
h-index

21
g-index

55
ext. papers

675
ext. citations

2.5
avg, IF

4.02
L-index

#	Paper	IF	Citations
44	Auditing buffer overflow vulnerabilities using hybrid staticdynamic analysis. <i>IET Software</i> , 2016 , 10, 54-61	1	6
43	Web Application Vulnerability Prediction Using Hybrid Program Analysis and Machine Learning. <i>IEEE Transactions on Dependable and Secure Computing</i> , 2015 , 12, 688-707	3.9	51
42	Buffer Overflow Vulnerability Prediction from x86 Executables Using Static Analysis and Machine Learning 2015 ,		7
41	Faceted Bug Report Search with Topic Model 2014 ,		1
40	Predicting Buffer Overflow Vulnerabilities through Mining Light-Weight Static Code Attributes 2014 ,		3
39	Detecting infeasible branches based on code patterns 2014 ,		3
38	Predicting SQL injection and cross site scripting vulnerabilities through mining input sanitization patterns. <i>Information and Software Technology</i> , 2013 , 55, 1767-1780	3.4	37
37	Binary Code Analysis. <i>Computer</i> , 2013 , 46, 60-68	1.6	6
36	Has this bug been reported? 2013 ,		6
35	Mining SQL injection and cross site scripting vulnerabilities using hybrid program analysis 2013 ,		45
34	Supporting the adaptation of open-source database applications through extracting data lifecycles. <i>IET Software</i> , 2013 , 7, 213-221	1	
33	Detection of Infeasible Paths: Approaches and Challenges. <i>Communications in Computer and Information Science</i> , 2013 , 64-78	0.3	5
32	Aiding Maintenance of Database Applications Through Extracting Attribute Dependency Graph. <i>Journal of Database Management</i> , 2013 , 24, 20-35	2.2	3
31	Automated Insertion of Exception Handling for Key and Referential Constraints. <i>Journal of Database Management</i> , 2013 , 24, 1-19	2.2	
30	Automated removal of cross site scripting vulnerabilities in web applications. <i>Information and Software Technology</i> , 2012 , 54, 467-478	3.4	45
29	Defending against Cross-Site Scripting Attacks. <i>Computer</i> , 2012 , 45, 55-62	1.6	18
28	Mining input sanitization patterns for predicting SQL injection and cross site scripting vulnerabilities 2012 ,		21

27	Detection of Buffer Overflow Vulnerabilities in C/C++ with Pattern Based Limited Symbolic Evaluation 2012 ,		1
26	Automated verification and testing of user-interactive undo features in database applications. <i>Software Testing Verification and Reliability</i> , 2012 , 22, 245-265	0.9	
25	Has this bug been reported? 2012 ,		9
24	Automated Insertion of Exception Handling for Key and Referential Constraints 2012 ,		2
23	Checking enforcement of integrity constraints in database applications based on code patterns. <i>Journal of Systems and Software</i> , 2011 , 84, 2253-2264	3.3	11
22	Defending against Buffer-Overflow Vulnerabilities. <i>Computer</i> , 2011 , 44, 53-60	1.6	4
21	Extraction of Attribute Dependency Graph from Database Applications 2011 ,		6
20	Measuring design complexity of semantic web ontologies. <i>Journal of Systems and Software</i> , 2010 , 83, 803-814	3.3	70
19	Covering code behavior on input validation in functional testing. <i>Information and Software Technology</i> , 2009 , 51, 546-553	3.4	20
18	Testing input validation in Web applications through automated model recovery. <i>Journal of Systems and Software</i> , 2008 , 81, 222-233	3.3	21
17	Applying static analysis for automated extraction of database interactions in web applications. <i>Information and Software Technology</i> , 2008 , 50, 160-175	3.4	18
16	An approach for the maintenance of input validation. <i>Information and Software Technology</i> , 2008 , 50, 449-461	3.4	7
15	Heuristics-based infeasible path detection for dynamic test data generation. <i>Information and Software Technology</i> , 2008 , 50, 641-655	3.4	33
14	Empirical-based recovery and maintenance of input error-correction features. <i>Journal of Software: Evolution and Process</i> , 2007 , 19, 419-450		1
13	An Empirical Study of Class Sizes for Large Java Systems. <i>Proceedings of the Asia Pacific Software Engineering Conference</i> , 2007 ,		6
12	An Empirical Study of Class Sizes for Large Java Systems 2007 ,		8
11	Automated verification and test case generation for input validation 2006 ,		7
10	An Approach to Aid the Understanding and Maintenance of Input Validation. <i>Conference on Software Maintenance, Proceedings of the</i> , 2006 ,		4

9	Automated elicitation of functional dependencies from source codes of database transactions. <i>Information and Software Technology</i> , 2004 , 46, 109-117	3-4	11
8	Automated elicitation of inclusion dependencies from the source code for database transactions. <i>Journal of Software: Evolution and Process</i> , 2003 , 15, 379-392		4
7	On formalization of the whole-part relationship in the Unified Modeling Language. <i>IEEE Transactions on Software Engineering</i> , 2003 , 29, 1054-1055	3-5	4
6	Software cost estimation through conceptual requirement 2003 ,		4
5	Exploring into programs for the recovery of data dependencies designed. <i>IEEE Transactions on Knowledge and Data Engineering</i> , 2002 , 14, 825-835	4-2	7
4	An approach for extracting code fragments that implement functionality from source programs. <i>Journal of Software: Evolution and Process</i> , 2001 , 13, 53-75		4
3	Reuse of components in data-intensive business programs through interface separation. <i>Information and Software Technology</i> , 1998 , 40, 15-25	3-4	
2	A method for the recovery of inclusion dependencies from data-intensive business programs. <i>Information and Software Technology</i> , 1997 , 39, 27-34	3-4	1
1	Components reuse for data-intensive business programs through an object-oriented architecture. <i>Journal of Systems and Software</i> , 1996 , 34, 3-20	3-3	1