Pettai, Martin; **Laud, Peeter**

Securing the future – an information flow analysis of a distributed OO language. (English)

Zbl 1302.68051

Bieliková, Mária (ed.) et al., SOFSEM 2012: Theory and practice of computer science. 38th conference on current trends in theory and practice of computer science, Špindlerův Mlýn, Czech Republic, January 21–27, 2012. Proceedings. Berlin: Springer (ISBN 978-3-642-27659-0/pbk). Lecture Notes in Computer Science 7147, 576-587 (2012).

Summary: We present an information-flow type system for a distributed object-oriented language with active objects, asynchronous method calls and futures. The variables of the program are classified as high and low. We allow while cycles with high guards to be used but only if they are not followed (directly or through synchronization) by an assignment to a low variable. To ensure the security of synchronization, we use a high and a low lock for each concurrent object group (cog). In some cases, we must allow a high lock held by one task to be overtaken by another, if the former is about to make a low side effect but the latter cannot make any low side effects. This is necessary to prevent synchronization depending on high variables from influencing the order of low side effects in different cogs. We prove a non-interference result for our type system.

For the entire collection see [Zbl 1236.68005].

MSC:

68N15 Theory of programming languages
68N19 Other programming paradigms (object-oriented, sequential, concurrent, automatic, etc.)

Software:

Creol; JFlow; JCobox

Full Text: DOI

References:

[1] 14th IEEE Computer Security Foundations Workshop (CSFW-14 2001), 11-13 Cape Breton, Nova Scotia, Canada. IEEE Computer Society (2001)

[2] Abadi, M.: Secrecy by Typing in Security Protocols. In: Ito, T., Abadi, M. (eds.) TACS 1997. LNCS, vol.˜1281, pp. 611–638. Springer, Heidelberg (1997)· doi:10.1007/BFb0014571

[3] Banerjee, A., Naumann, D.A.: Secure Information Flow and Pointer Confinement in a Java-like Language. In: CSFW, p. 253. IEEE Computer Society (2002)

[4] Barthe, G., Rezk, T.: Non-interference for a JVM-like language. In: Morrisett, J.G., Fähndrich, M. (eds.) TLDI, pp. 103–112. ACM (2005)· doi:10.1145/1040294.1040304

[5] Barthe, G., Rezk, T., Naumann, D.A.: Deriving an Information Flow Checker and Certifying Compiler for Java. In: IEEE Symposium on Security and Privacy, pp. 230–242. IEEE Computer Society (2006)· doi:10.1109/SP.2006.13

[6] Bernardeschi, C., De Francesco, N., Lettieri, G.: Concrete and Abstract Semantics to Check Secure Information Flow in Concurrent Programs. Fundamenta Informaticae˜60(1-4), 81–98 (2004) · Zbl 1083.68065

[7] Boudol, G., Castellani, I.: Noninterference for concurrent programs and thread systems. Theor. Comput. Sci.˜281(1-2), 109–130 (2002) · Zbl 0997.68022 · doi:10.1016/S0304-3975(02)00010-5

[8] de Boer, F.S., Clarke, D., Johnsen, E.B.: A Complete Guide to the Future. In: De Nicola, R. (ed.) ESOP 2007. LNCS, vol.˜4421, pp. 316–330. Springer, Heidelberg (2007) · Zbl 05187163 · doi:10.1007/978-3-540-71316-6$_2$2

[9] Goguen, J.A., Meseguer, J.: Security Policies and Security Models. In: IEEE Symposium on Security and Privacy, pp. 11–20 (1982)· doi:10.1109/SP.1982.10014

[10] Hähnle, R., Johnsen, E.B., Østvold, B.M., Schäfer, J., Steffen, M., Torjusen, A.B.: Report on the Core ABS Language and Methodology: Part A. Highly Adaptable and Trustworthy Software using Formal Models (HATS), Deliverable D1.1A˜4 (2010)

[11] Honda, K., Vasconcelos, V.T., Yoshida, N.: Secure Information Flow as Typed Process Behaviour. In: Smolka, G. (ed.) ESOP 2000. LNCS, vol.˜1782, pp. 180–199. Springer, Heidelberg (2000) · Zbl 0960.68126 · doi:10.1007/3-540-46425-5$_1$2

[12] Johnsen, E.B., Blanchette, J.C., Kyas, M., Owe, O.: Intra-Object versus Inter-Object: Concurrency and Reasoning in Creol. Electr. Notes Theor. Comput. Sci.˜243, 89–103 (2009)· doi:10.1016/j.entcs.2009.07.007

[13] Mantel, H., Sabelfeld, A.: A Generic Approach to the Security of Multi-Threaded Programs. In: CSFW [1], p. 126 · Zbl

1015.68509 · doi:10.1109/CSFW.2001.930142

[14] Myers, A.C.: JFlow: Practical Mostly-Static Information Flow Control. In: POPL, pp. 228–241 (1999)· doi:10.1145/292540.292561

[15] Pettai, M., Laud, P.: Securing the Future – an Information Flow Analysis of a Distributed OO Language. Technical Report T-4-14, Cybernetica AS (2011) · Zbl 1302.68051

[16] Russo, A., Hughes, J., Naumann, J.D.A., Sabelfeld, A.: Closing Internal Timing Channels by Transformation. In: Okada, M., Satoh, I. (eds.) ASIAN 2006. LNCS, vol.˜4435, pp. 120–135. Springer, Heidelberg (2008) · Zbl 05252968 · doi:10.1007/978-3-540-77505-$8_1$0

[17] Russo, A., Sabelfeld, A.: Security for Multithreaded Programs Under Cooperative Scheduling. In: Virbitskaite, I., Voronkov, A. (eds.) PSI 2006. LNCS, vol.˜4378, pp. 474–480. Springer, Heidelberg (2007) · Zbl 1185.68230 · doi:10.1007/978-3-540-70881-$0_4$3

[18] Sabelfeld, A.: Confidentiality for Multithreaded Programs via Bisimulation. In: Broy, M., Zamulin, A.V. (eds.) PSI 2003. LNCS, vol.˜2890, pp. 260–274. Springer, Heidelberg (2004) · Zbl 1254.68087 · doi:10.1007/978-3-540-39866-$0_2$7

[19] Sabelfeld, A., Mantel, H.: Securing Communication in a Concurrent Language. In: Hermenegildo, M.V., Puebla, G. (eds.) SAS 2002. LNCS, vol.˜2477, pp. 376–394. Springer, Heidelberg (2002)· doi:10.1007/3-540-45789-$5_2$7

[20] Sabelfeld, A., Sands, D.: Probabilistic Noninterference for Multi-Threaded Programs. In: CSFW, pp. 200–214 (2000)· doi:10.1109/CSFW.2000.85693

[21] Schäfer, J., Poetzsch-Heffter, A.: JCoBox: Generalizing Active Objects to Concurrent Components. In: D'Hondt, T. (ed.) ECOOP 2010. LNCS, vol.˜6183, pp. 275–299. Springer, Heidelberg (2010) · Zbl 05773898 · doi:10.1007/978-3-642-14107-$2_1$3

[22] Smith, G.: A New Type System for Secure Information Flow. In: CSFW [1], pp. 115–125· doi:10.1109/CSFW.2001.930141

[23] Smith, G.: Probabilistic Noninterference through Weak Probabilistic Bisimulation. In: CSFW, pp. 3–13. IEEE Computer Society (2003)· doi:10.1109/CSFW.2003.1212701

[24] Smith, G., Volpano, D.M.: Secure Information Flow in a Multi-Threaded Imperative Language. In: POPL, pp. 355–364 (1998)· doi:10.1145/268946.268975

[25] Volpano, D.M., Irvine, C.E., Smith, G.: A Sound Type System for Secure Flow Analysis. Journal of Computer Security˜4(2/3), 167–188 (1996) · Zbl 05430401 · doi:10.3233/JCS-1996-42-304

[26] Zheng, L., Chong, S., Myers, A.C., Zdancewic, S.: Using Replication and Partitioning to Build Secure Distributed Systems. In: IEEE Symposium on Security and Privacy, pp. 236–250. IEEE Computer Society (2003)