International Journal of Engineering& Scientific Research
Vol. 6 Issue 7, July 2018,
ISSN: 2347-6532 Impact Factor: 6.660
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

An Enhanced Reusable Software Framework to Select Secure and Scalable Component (ERSMSS-SC)

Kiranpal Singh Virk[1], Dalip[2]
[1] Assistant Professor, Department of Computer Science, Guru Nanak Khalsa College, Yamuna Nagar, Haryana 135001, India
{kiranpal.virk}@yahoo.com
[2] M.M. Institute of Computer Technology & Business Management
Maharishi Markandeshwar Deemed to be University, Mullana (Ambala), Haryana, India
{dalipkamboj}@mmumullana.org

ABSTRACT

To validate the components that provides desired functionality in terms of security and scalability), from the finite set of component set by the use of defined process for software component selection. The selection process helps in choosing the optimal set of components from the third party repository. In order to select the optimal set of component having multiple attributes (Se, Sc), our technique selects best component, according to our criterion, among the available components for the same purpose. Many techniques have been proposed for component selection under varied situations to select the optimal component from component set of same functionality. This paper proposes framework for component selection after going through a brief survey of the component selection techniques.

Keywords: reusable component, CBSD, security, scalability, metrics**,** software components, software development, software engineering, software quality, framework based component selection process.

1. INTRODUCTION

Recent technology revolution of Embedded System has churned a new paradigm i.e. Internet of Things (IoT). IoT has brought technology to the door of the mankind. It has provided various verticals of applications. With the advent of IoT or Cyber Physical Systems and other latest technologies, the reusability has entered into a new domain. The

plug and play option has reached to the new level. Need based change of hardware devices (sensors, actuators) or software components could be replaced with newer version. the need to secure the applications running on such systems gained the focus. Fast of IoT platform based applications in smart homes, offices, cars, cities, garbage disposal, health management, agriculture, flora and fauna management has given way to security concern as failure of an IoT application directly affects the human life and environment. In order to deliver fast, the developers and the organization tend to compromise on the basic principles of Component Based Software Engineering. This paper is an effort to review the work done regarding the security threats, scalability issues and their mitigation on IoT application from the point of component reusability and suggest a design which evaluates the Security and Scalability levels of a system with methodology of multi-metric approach.

Consider when a system is build from scratch using the traditional established life cycle models, user context may change considerably at the final stages. Under these circumstances only two options are left. First being to scrap the project, which is not viable in today's profit conscious industry. Second one is two explore the third party reliable and reusable software components or COTS. The process of building software systems by assembling and integrating third party software components has become a strategic need in a wide variety of application areas. A software system may include one or more COTS components (products). If some requirement(s) cannot be satisfied with COTS components, then the component(s) corresponding to the given system requirement(s) may be developed in-house.

IoT applications can be considered as Service-oriented Component-based Applications that takes this scenario of component reusability a step further. Service-oriented Component-based Applications provides a framework to construct modularised applications consisting of software components that uses software services provided by other components. It is this reusability that poses a risk of an unidentified nature. Existing IoT application is easily extendable by adding new sensors and accordingly adding new code in the shape of a reusable software component as COTS. The reusable component added this way may expose the existing applications from within for an attack from outside. **The internal risks** are risk arising due to design faults or implementation errors like code errors. Normally a component developed using Component Based Software Engineering principle would have well defined input and output interactions. Such interactions help in measuring the cohesion and coupling metrics of a component. Manadhata et. al. in their work have considered Input/Output automata of a component to define its entry and exit points[1].

The empirical study conducted by Grechanik et. al. [3] suggested that majority of the interactions occur within the defined security boundaries of an application. And the topologies of the security measures and component pattern interactions were developed to suggest an architecture. **The external risks** covers the risk arising from individual unattended ES devices, communication between ES devices, information backyards like cloud databases. In most of the secured systems wherein the ES device is lying unattended on the pretext that there is a 'air gap', the security has been compromised by various attacks like Stuxnet worm attack in 2010. Mirai attack of 2016 is another indicator that in hastiness of implementation of technology wherein the remote ports with default username and password are left open to be exploited later on by the hackers. The works like [4][5] focussed upon the external risks and their mitigation by achieving access control (authorization and authentication) between a cyber physical device and the cloud storage with end-to-end communication security

2. LITERATURE SURVEY

According to Lichota et al. [20] portable, reusable, integrated, software module (PRISM) consists of generic component architecture and a process of integrating a product as a component of this architecture. In PRISM there are different phases of product evaluation process like identification, screening, stand alone test, integration test and field

Haining et.al. presented, "Towards A Semantic based Approach for Software Reusable Component Classification and Retrieval", an approach for classification and retrieval of the software components [19].

Manadhata and Wing [6][7]further suggested approach for enhancing security level by categorizing the approaches as system-centric approach and attack centric approach.

In attack-centric approach, factors like behaviour, resources and capabilities of the attackers that lead to vulnerability risks. In system-centric approach system design and configurations forms the core focus area.

Various authors have suggested a notion of system's attack surface using input/output or entry/exit points of a component using Input/ Output automata model or similar techniques [1][8][9][10]. These works have considered the attack surface of software as base criteria for evaluating the security. From a set of system's resources, a sub set called system's attack surface is defined. Reducing the attack surface is one of the ways for making software more secure. Attackers exploits the system's resources like system's methods (API), channels (sockets), and data items (input streams) for attacking sandboxes)

Al-Sarayreh and Abran suggested the use of and the COSMIC generic software model suggests the use of data movement for Entry, Exit, Write and Read for measuring function size of components in a business application with humans and another 'peer' application as its functional users[11] [12].

Abran and Soubra [13] implemented the COSMIC approach on IoT application using Arduino open source. They suggested how Entry, Exit, Write and Read points are indentified and then calculated Cosmic Function Points that could be helpful in ensuring optimum battery load for continuity and quality of service in energy constrained IoT frameworks.

Multiple works[14][15][16][17] by Josef Noll suggested the multi-metric approach for measuring security, privacy and dependability in a complex system and suggested an implementation on a Smart Grid on the footprints of cyber physical systems or IoT paradigm

Feature models (Kang et al. 1990) are a simple but powerful formalism for representing commonalities, varying aspects, and configuration rules of software products.

In (Moisan et al. 2011), feature models were proposed for the representation and dynamic adaptation of component-based systems, such as a video surveillance (VS) processing. chain. The domain of computer vision and video surveillance offers a challenging ground because of the high variability in both the surveillance tasks and the video analysis algorithms.

From an implementation perspective, selecting the (software) components themselves, assembling them, and tuning their parameters to comply with the context might lead to different configuration variants. The global properties of the system are computed by means of aggregate functions over the features.

In (Sanchez et al. 2013), we presented a heuristic search algorithm called CSA (Configuration Selection Algorithm) for solving the optimization problem resulting from selecting a valid configuration of a system based on feature models.

3. COMPONENT SELECTION MODEL

Based upon the above discussion, a model could be worked upon wherein the reusable component is evaluated before its inclusion in an existing application from security point of view. The system-centric approach or the internal threat approach could be further worked upon along with COSMIC generic software model to suggest a working model for the evaluation of security and scalability.

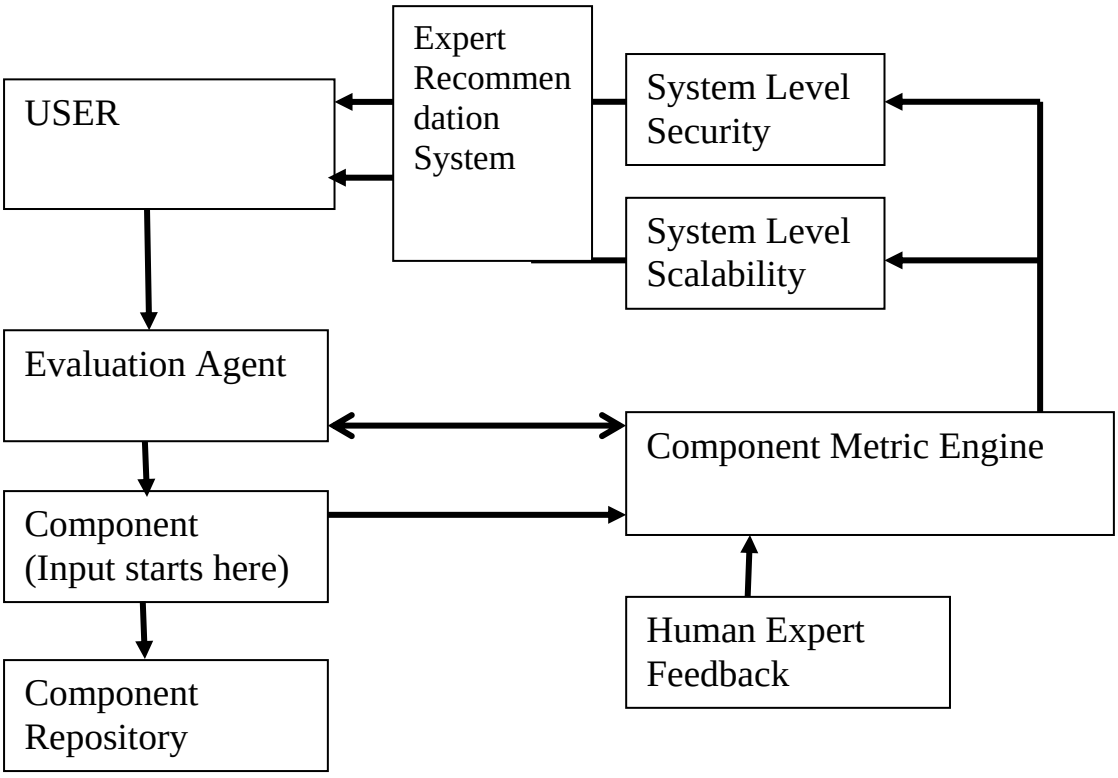

Figure 3.1: working model for the evaluation of security and scalability.

3.1 METHODOLOGY

This section describes the methodology to measure the Security and Scalability (SeSc) level of a system. The objective is to achieve an overall system $SeSc_{level}$, $SeSc_{system}$. The main advantage of this methodology is that it provides a simple mechanism to measure and evaluate the system security and scalability levels. $SeSc_{system}$ is a doublet, composed of individual Security and Scalability levels (se,sc). Each of the levels is represented by a range between 0 and 100, i.e. the higher the number, the higher the Security and Scalability level. However, in order to end up with $SeSc_{system}$, during the whole process, the criticality is evaluated. Criticality is again a couplet (Cse,Csc, defined as the complement of SeSc, and expressed as (Cse,Csc = (100, 100) − (se,sc). Direct assigned attributes/Quantitative Attributes/Qualitative attributes have properties for security and scalability which further have combination of selected/Unselected/Deselected properties depending upon user

preference or context change like criticality of human life is different in ultrasonic sensor used in high speed car or used in blind assisting devices

4 CONCLUSION

This paper presents a methodology for assessing security and scalability of embedded systems. Embedded Systems evolved from isolated to highly interconnected devices, becoming the key elements of the Internet of Things. Our approach combines the assessment of security and scalability, thus allows the optimisation towards a balanced solution. In order to address the challenge of a balanced solution, the Multi-Metrics methodology presented in this paper considers all security and scalability aspects together. The main advantages of the methodology are the simplicity, Multi-Metrics is the core process used along all the steps, and scalability, it starts with component evaluation to jump over sub-systems and ends up with the entire system evaluation. The result is an overall $SeSc_{System}$ level, which makes it easy to understand under which configuration the system will perform as envisaged by the $SeSc_{Goal}$. The paper analyses a total of 01 configuration, and concentrates on the ultrasonic sensor.

REFERENCES:

1. Manadhata, P. K., Kaynar, D. K. and Wing, J. M. (2007) 'A Formal Model for a System's Attack Surface'. doi: 10.21236/ADA477014.
2. Karati, A., Amin, R., Islam, S. K. H. and Choo, K. K. R. (2018) 'Provably secure and lightweight identity-based authenticated data sharing protocol for cyber-physical cloud environment', IEEE Transactions on Cloud Computing, pp. 1–14. doi: 10.1109/TCC.2018.2834405.
3. Grechanik, M., Perry, D. E. and Batory, D. (2006) 'A security mechanism for component-based systems', Proceedings - Fifth International Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems, 2006, pp. 53–62. doi: 10.1109/ICCBSS.2006.3.
4. Karati, A., Amin, R., Islam, S. K. H. and Choo, K. K. R. (2018) 'Provably secure and lightweight identity-based authenticated data sharing protocol for cyber-physical cloud environment', IEEE Transactions on Cloud Computing, pp. 1–14. doi: 10.1109/TCC.2018.2834405.
5. Ragunathan R. (2012) 'A cyber–physical future', Proceedings of the IEEE, 100(Special Centennial Issue):1309–1312.

6. P. K. Manadhata and J. M. Wing (2011) 'An attack surface metric', IEEE Transactions on Software Engineering, vol. 37, no. 3, pp. 371–386.

7. R. Yesudas and R. Clarke (2013) 'A framework for risk analysis in smart grid', International Workshop on Critical Information Infrastructures Security. Springer, pp. 84–95.

8. Howard M., Pincus J., Wing J.M. (2005) 'Measuring Relative Attack Surfaces', In: Lee D.T., Shieh S.P., Tygar J.D. (eds) Computer Security in the 21st Century. Springer, Boston, MA. doi: 10.1007/0-387-24006-3_8

9. J. Szefer, E. Keller, R. B. Lee, and J. Rexford(2011) 'Eliminating the hypervisor attack surface for a more secure cloud', in Proceedings of the 18th ACM Conference on Computer and Communications Security, ser. CCS '11. ACM, pp. 401–412.

10. A. Bartel, J. Klein, Y. Le Traon, and M. Monperrus (2012) 'Automatically securing permission-based software by reducing the attack surface: An application to android', Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering, ser. ASE 2012. ACM, pp. 274–277.

11. Al-Sarayreh, K. T. and Abran, A. (2010) 'A generic model for the specification of software interface requirements and measurement of their functional size', 8th ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2010, (October 2016), pp. 217–222. doi: 10.1109/SERA.2010.35.

12. COSMIC Measurement Manual Version 4.0.1, Common Software Measurement International Consortium, 2015.http://cosmic-sizing.org

13. Soubra H. and Abran A. (2017) ' Functional size measurement for the internet of things (IoT): an example using COSMIC and the Arduino open-source platform' , Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement (IWSM Mensura '17). ACM, New York, NY, USA, 122-128. DOI: ttps://doi.org/10.1145/3143434.3143452

14. Noll, J., Garitano, I., Fayyad, S., Asberg, E. and Abie, H. (2015) 'Measurable Security, Privacy and Dependability in Smart Grids', Journal of Cyber Security and Mobility. doi: 10.13052/jcsm2245-1439.342.

15. Fayyad, S. and Noll, J. (no date) 'Toward Objective Security Measurability and Manageability'. doi: 10.1109/HONET.2017.8102211.

16. Garitano, I., Fayyad, S. and Noll, J. (2015) 'Multi-Metrics Approach for Security, Privacy and Dependability in Embedded Systems', Wireless Personal Communications. doi: 10.1007/s11277-015-2478-z.

17. Fayyad, S. and Noll, J. (2017) 'A framework for measurability of security', 2017 8th International Conference on Information and Communication Systems, ICICS 2017, (April), pp. 302–309. doi: 10.1109/IACS.2017.7921989.

18. Emiliano Sanchez, L. et al. (2011) 'An approach based on feature models and quality criteria for adapting component-based systems', Journal of Software Engineering Research and Development, 3(10), pp. 1–30. doi: 10.1186/s40411-015-0022-1.

19. Haining Yao , Letha Etzkorn, "Towards A Semantic-based Approach for Software Reusable Component Classification and Retrieval", ACM Southeast Regional Conference, 2004

20. Lichota, R.W., Vesprini, R.L. and Swanson, B. (1997) 'PRISM: product examination process for component based development', Proc. of SAST '97, pp.61–69.