Securing data access and automatic exchanges System

Phadatare Manasi Mahadeo, Dr. T. V. Gopal, Raj Bahadur Singh
Ph.D Research Scholar: SRM Institute of Science & Technology (SRMIST), SRM University Chennai India.
Dean – CET, Kattankulathur SRM Institute of Science & Technology (SRMIST), SRM University Chennai India.
Co-Guide: Prof.(Dr.) Ashoka Institute of Technology and Management, Varanasi, Uttar Pradesh 221007

Abstract:
When evaluating project regressions, our invention, "Securing data access and automatic exchanges System" has become a key agenda item for managing software upgrades that need the least amount of manual labor. For software development methodologies like Agile, Scrum, and XP, rapid testing environments require continuous integration solutions. The main problem is that several technologies are used since no one solution can handle project automation. The recommended automation tool need to provide debugging, execution, and configuration features. Project automation systems like Jenkins and Apache Continuum for task scheduling, Selenium and Testing for test management, and Mercurial and Get for software configuration management are hard to combine. The difficulty increases when an organization wishes to use the present cloud services since data across software tools and processes is not shared by the PKI infrastructure for access control that is already in place. The proposed technique takes a single CSV containing input test case and metadata information, and uses it to categorize and perform the tests automatically. The recommended method requires security access control methods to be included in the platform used for task execution in a cloud environment.
Keywords: Application, Securing, data, access, automatic, exchanges
DOI: 10.24297/j.cims.2023.12.106

1. Introduction

We securely provide network telemetry data with authorized third parties, preserving privacy while allowing searches on the encrypted network telemetry. Using more advanced security algorithms like ID-PKC and ABE, we carried out an experimental inquiry into the creation of a search algorithm that protects privacy [1].In order to check for bugs in the developed software before releasing it, a fast development environment is required for the product that will be provided on the cloud platform. Fast software development demands the creation of an unbreakable infrastructure that is safe, scalable, and reliable [2].

However, even while cloud systems allow for the rapid development of products, they still need the assistance of automation tools in order to fully use the build and test environments [3]. With the help of cloud virtualization services, the suggested solution handles the issue of building and testing a framework in a cloud environment [4].

2. Literature Survey on Automation Frameworks and Secure Jobs Execution

Different automation technologies in the cloud environment are required for the various stages of software development. For instance, technologies like Puppet, Juju, Apache Continuum, Jenkins, and Cobbler are used in various phases to setup, schedule, and monitor automated jobs [33]. We need a solution that can do continuous integration, test-driven development, and debugging in order to increase quality and quickly resolve client concerns.

Based on the project requirement, automation can be triggered in various methods some of them contains the following (34).On-demand run: a user manually initiates the jobs using scripts or a user interface. Tests are organized into test suites and run in accordance with the schedule during a scheduled run. As soon as the build is accessible or a change set is found (pushed) in the source repository, these tests will begin to execute. (iii)On event occurrence. A plan was developed to assess the Cloud Service Level Agreement (SLA) in a real-time context, and a security management system was suggested for keeping track of alerts, security events involving the cloud, and vulnerabilities [5]. Google Colas, a component of the Google App Engine, serves as the main run-time environment [6, 7].

Cloud computing represents the next step in the growth of the Internet. For coherence to be achieved throughout a network, it is necessary to share resources [38]. In recent years, it has developed as a new computing standard that has implications for a variety of academic domains, including software testing.

Proposed Approach

There are two main components to the suggested technique. First, we provide a framework for the execution of automated processes in order to thoroughly test and validate cloud software projects. The new code change tests are essential because they enable any project to work with newly changed source code without regressing. We provide an all-inclusive methodology that

scales and executes tests utilizing cloud resources and in accordance with configuration (serial or parallel), respectively.

The suggested method effectively groups and performs the tests automatically by using a single CSV (comma separated file) with input test case and metadata information.

A CSV (comma separated) file contains the test configuration data.
Later, we propose a security access control mechanism for the jobs execution platform leveraging the cloud environment. The role-based access control policies are integrated, and the security aspects of the tool are experimented.

The test information listed under each item in the test definition (CSV) file include the following: UNIQUE_ID, TEST_NAME, EXECUTION_MODE, and MACHINE_CONFIG...

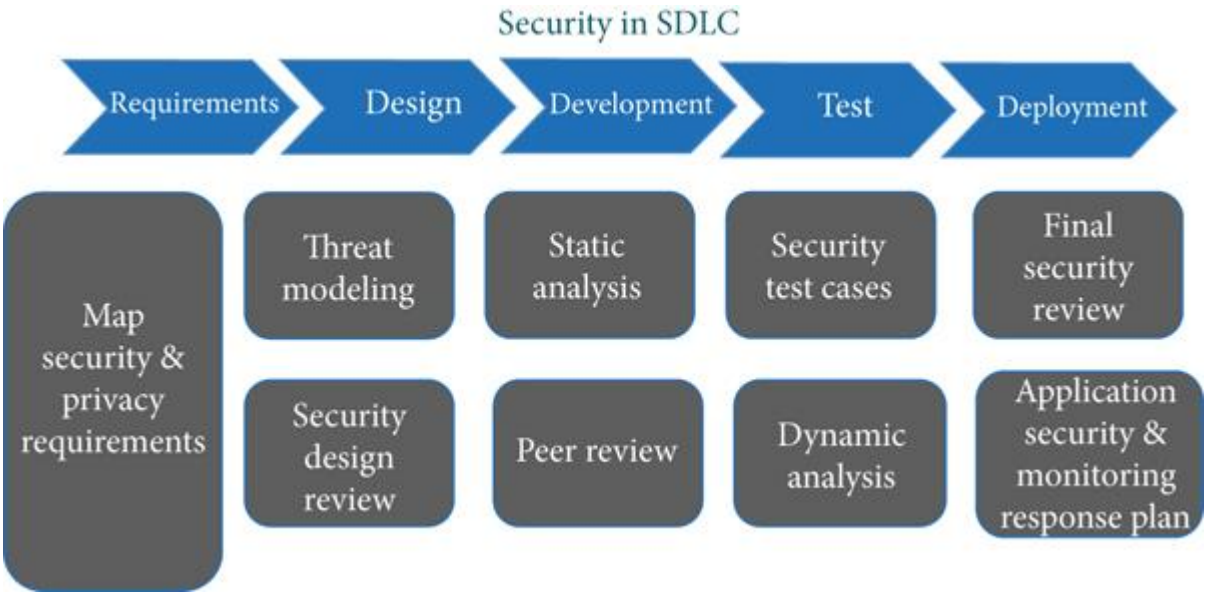

Fig.1: Application of Securing data access and automatic exchanges Flow

Design and Implementation of Fully Automated Test Framework for Cloud

For the development and deployment of software for a cloud environment, extra controls are necessary in contrast to on-premise installations. Compatibility problems resulting from cloud providers' support for various methods of managing user data, a reduction in control over the physical security of the cloud, compliance with relevant laws that apply to the data domain, and data security throughout the entire tenure of the data (newly created persisted, processed, transited, and destroyed) are just a few of the significant changes in the cloud environment.

As seen in Figure 1, cloud software developers should adhere to the Secure Coding Guidelines (OWASP Security Guide). Every stage, from specifications through deployment, must take security into account. The execution logs, debug statements, and error messages all include extremely useful information about the code in the production environment. Static analysis checks code statically, and the dynamic code analysis tools analyze as it executes and checks for possible security vulnerabilities. The dynamic code analysis tool analyzes the runtime code execution paths.

Experimental Results

The user-interface tests are picked in order to experiment and validate the outcomes of the automation framework. In general, the tests are frequently divided into user-interface, database, and application programming interface tests. Due to the variety of browsers and operating systems required, the support for each operating system, the need for pre- and post-configuration of virtual machines, and the runtime environment, user-interface tests are said to be the most complicated. Execution of this test requires careful consideration of the setup of the target computer, and therefore takes more time than other tests. The development of the execution time was virtually linear with the quantity of test cases.

Table 1 : Test case results using Single client machine.

No. of test cases	Execution time (in minutes)
40	86
70	112
100	170
150	292
200	387
250	458
300	503

3. Result

They only care about metrics and procedures for assessing scalability in parallel and distributed systems when it comes to performance testing. Aspects like dynamic scalability are not supported in the stage that metrics development, frameworks, and solutions are now in. Software issues and bug patches affect regression testing, which raises concerns and presents challenges. The on-demand cloud testing services should be able to address a wide range of issues and worries. Test engineers should be given enough test models and criteria that work well with cloud computing in order to provide suitable test models and criteria for cloud testing. As a continuous validation and regression testing solution, test engineers must provide automated retesting approaches that address the multitenancy component of cloud computing environments whenever software has been upgraded as a consequence of bug patches or feature upgrades. Since both cloud and Saabs apps offer connection protocols and APIs, test engineers should verify the interoperability quality of cloud applications. For cloud interoperability, new automated test options are available.

4. Conclusion

Flexible infrastructure is necessary for quick software product development. Such products may be created by utilizing the elasticity offered by cloud environments. To generate quick products, the current cloud environment has to include automation of the build and test environments. When it comes to cloud computing and data calculations, security comes first. This technique has as its goal the development of a plan for a safe cloud environment. this approach covered the necessity for data encryption while using public cloud services for compute, security forensics, and auditing. The importance of a safe SDLC for scalable test automation frameworks built on top of cloud environments was also underlined by this work. Functional testing is growing increasingly advanced and uses a lot of hardware and software to mimic human activity. In contrast to functional testing, nonfunctional testing enables measurement and association testing of software systems' nonfunctional features. According to experts, cloud computing has a limited number of advantages and testing problems. Every project must develop and record new requirements since testing is an ongoing effort.

References

1. K. Beck, M. Beedle, A. Van Bennekum et al., *Manifesto for Agile Software Development*, Agile Alliance, 2022.

2. P. Samarati, "Data Security and Privacy in the Cloud," *Information security practice and experience*, Springer, Cham, 2022.
3. P. Donadio, G. B. Fioccola, R. Canonico, and G. Ventre, "Network security for hybrid cloud," in *2014 Euro Med Telco Conference (EMTC)*, pp. 1–6, Naples, Italy, 2021.
4. N. K. Rathore, N. K. Jain, P. K. Shukla, U. S. Rawat, and R. Dubey, "Image forgery detection using singular value decomposition with some attacks," *National Academy Science Letters*, vol. 44, no. 4, pp. 331–338, 2021.
5. N. Garigipati and R. V. Krishna, "A study on data security and query privacy in cloud," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 337–341, Tirunelveli, India, 2019.
6. P. K. Shukla and M. Dixit, "Cloud-based image fusion using guided filtering," *Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing*, IGI Global, Hershey, PA, pp. 146–165, 2016.
7. P. K. Shukla, V. Roy, P. K. Shukla et al., "An advanced EEG motion artifacts eradication algorithm," *The Computer Journal*, 2021, bxab170.
8. R. Buyya, C. Vecchiola, and S. T. Selvi, *Mastering Cloud Computing: Foundations and Applications Programming*, Newnes, 2013.
9. C. Yang, B. Song, Y. Ding, O. Jiangtao, and C. Fan, "Efficient data integrity auditing supporting provable data update for secure cloud storage," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 5721917, 12 pages, 2022.
10. G. G. Rajput and R. Chavan, "improved LSB based image steganography using run length encoding and random insertion technique for colour images," *World Scientific News*, vol. 112, pp. 180–192, 2018.
11. D. Parwani, A. Dutta, P. K. Shukla, and M. Tahiliyani, "Various techniques of DDoS attacks detection and prevention at cloud: a survey," *Oriental Journal of Computer Science and Technology*, vol. 8, no. 2, pp. 110–120, 2015.
12. A. Sarkar and S. Karforma, "A new pixel selection technique of LSB based steganography for data hiding," *International Research Journal of Computer Science (IRJCS)*, vol. 5, no. 3, pp. 120–125, 2018.
13. H. Deng, Z. Qin, Q. Wu et al., "Achieving fine-grained data sharing for hierarchical organizations in clouds," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 1, p. 1, 2022.

14. J. Shen, H. Yang, P. Vijayakumar, and N. Kumar, "A privacy-preserving and untraceable group data sharing scheme in cloud computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 114, 2021.
15. G. Khambra and P. Shukla, "Novel machine learning applications on fly ash based concrete: An overview," *Materials Today: Proceedings*, pp. 2214–7853, 2021.
16. K. Bahwaireth, L.' a. Tawalbeh, E. Benkhelifa, Y. Jararweh, and M. A. Tawalbeh, "Experimental comparison of simulation tools for efficient cloud and mobile cloud computing applications," *Journal on Information Security*, vol. 2016, no. 1, p. 15, 2016.
17. S. S. Kushwaha, S. Joshi, D. Singh, M. Kaur, and H.-N. Lee, "Systematic review of security vulnerabilities in Ethereum blockchain smart contract," *IEEE Access*, vol. 10, pp. 6605–6621, 2022.
18. G. Xu, M. Lai, J. Li, L. Sun, and X. Shi, "A generic integrity verification algorithm of version files for cloud deduplication data storage," *EURASIP Journal on Information Security*, vol. 12, no. 1, 2018.
19. S. Mehdi and F. Richard Yu, "Attribute-based data access control in mobile cloud computing: taxonomy and open issues," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 10, pp. 2991–3005, 2016.
20. C. Bernardini, M. R. Asghar, and B. Crispo, "Security and privacy in vehicular communications: challenges and opportunities," *Vehicular Communications*, vol. 10, pp. 13–28, 2017.