

Secure Online Judge in Cloud Environment

Himanshu Sharma¹, Rachna Jain², Siddhant Shekhar³, Aabhas Sinha⁴

¹Computer Science Department Bharati Vidyapeeth's College of Engineering New Delhi, India

²Computer Science Department Bharati Vidyapeeth's College of Engineering New Delhi, India

³Computer Science Department Bharati Vidyapeeth's College of Engineering New Delhi, India

⁴Computer Science Department Bharati Vidyapeeth's College of Engineering New Delhi, India

Abstract: Cloud Computing is a rapidly developing technology in the IT Sector. It provides an on demand service to its users using the Internet. It provides the users with the resources as per their demand and also enables the users to pay as per their usage of resources. The users can access the cloud anytime using the Internet. Such features eliminate the requirement of carrying documentation for the users as the cloud can be accessed globally and remotely by the user. But, this introduces the possibility of security breaches or intruder attacks while the user interacts with the cloud database. In this paper, some of the security concerns while using a Cloud based Online Judge will be addressed using new approaches. An Online Judge is a system which takes a code, compiles it, checks for compilation errors, and executes the code by giving it some hidden inputs and capturing the output produced after it. It checks for runtime errors and finally matches the output with the expected output from the solution. The Online Judge system is very important for grading of coding assignments and for programming contests as it can include the constraints on the solutions for source code limit and various resources such as time and space requirements. It also removes the subjective nature in the grading system of coding assignments. The Online judge faces various security issues at authorization level and access rights of users. There is also possibility of a user running a malicious code on the judge thus harming the server or other resources. In this paper we address such issues.

Keywords: Online Judge, Security, Cloud computing, Code checking.

I. Introduction

Cloud Computing is considered as an emerging field due to the provision of “on demand service” to its users via the Internet. The goal of cloud computing is to provide the users with resources as per their requests of the service. It is an adept way to share or use resources as it provides services to people remotely while letting users get the resources as per their purpose. Cloud services can be attained on three different layers: Platform as a Service (PaaS), Infrastructure as a Service (IaaS) and Software as a Service (SaaS).

Cloud Computing consists of three different deployment methods:

Private cloud is a type of cloud computing that delivers similar advantages to public cloud, including scalability and self-service, but through a proprietary architecture. Unlike public clouds, which deliver services to multiple organizations, a private cloud is dedicated to a single organization. A public cloud is one in which the services and infrastructure are provided off-site over the Internet. These clouds offer the greatest level of efficiency in shared resources; however, they are also more vulnerable than private clouds. Hybrid cloud is a cloud computing environment which uses a mix of on-premises, private cloud and third-party, public cloud services with orchestration between the two platforms.

Section 2 elaborates the data gathered via the various related work done in the fields of online judges and its applications and cloud computing environment security issues, algorithms and management.

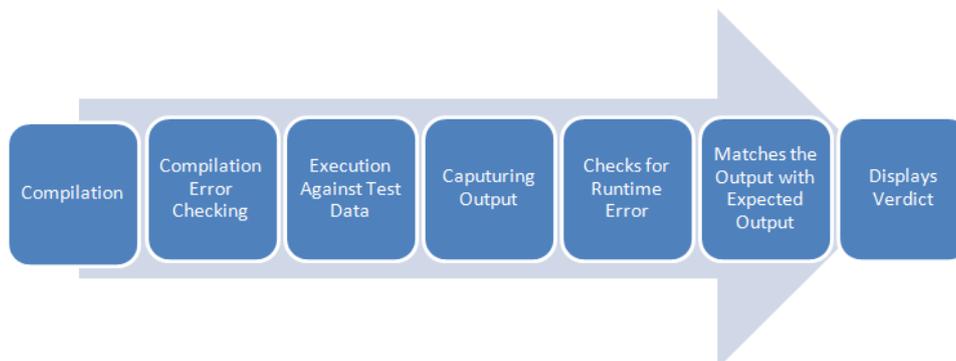


Fig. 1 Process involved in Online Judge

The security issues and concerns are described and argued upon in section 3 for an Online Judge in a cloud computing environment. These issues elaborately discuss the possible ways the cloud environment can be breached. The implementation details for the proposed Online Judge are discussed in section 4 which describes the script details for the working of the software as well as basic processes and tasks that the Online Judge or its working subprograms are to perform.

An Online Judge is an online system deployed in a programming community which can take a code, compiles it, checks for compilation errors, and executes the code by giving it some hidden inputs and capturing the output produced after it. It checks for runtime errors and finally matches the output with the expected output from the solution.

An Online Judge can be used for grading the coding assignments in institutions as it removes the subjective nature of grading in the manual process. As almost all of the computer science courses require programming, it becomes necessary to have an automatic grading and checking system for the solutions. Since, the advanced problems involving algorithms and data structures cannot be tested manually on large input data, automatic grading comes into play.

The manual grading has its own disadvantages such as subjective nature of the grading. For example, if a user solves a problem correctly using some other approach unknown to the evaluator, he may not accept that solution. On the other hand, if someone solves the problem that resembles the solution used by the evaluator, with some bug in the source, he may accept it.

There is another possibility of inefficient programs being accepted by the evaluator as he may not be able to test the solution on large test data for efficiency. There is no manual grading method that will be consistent for all the solutions to test the efficiency. With a form of online judge, we can set a time limit on the solutions such that if any program that has not been terminated by that time limit is considered inefficient and is not accepted by the judge. Another disadvantage of manual grading is that it is extremely slow. Most of the solutions in competitive programming setting are judged in a matter of few seconds, which is impossible with manual grading.

II. Related Work

Akashdeep Bhardwaja et al. in 2001 [1] discussed the various symmetric as well as asymmetric algorithms that can be applied so as to encrypt the program code that the users deem worthy of protection and want to keep hidden. Some of the asymmetric key encryption techniques discussed are RSA and Diffie Hellman and AES and Blowfish are some of the symmetric encryption algorithms. Andy Kurnia et al. [2] describes the applicability and implementation of the automatic program grading system known as Online Judge. This paper discusses the aesthetic aspects of Online Judge System.

The crucial aspects of cloud computing are characterized by Peter Mell and Timothy Grance [9] with further study of the various service models and deployment models in a cloud computing environment along with its essential characteristics. Wayne Jansen et al. [3] tells us about the public cloud services and key security and privacy issues such as compliance, trust, identity and access management and architecture in cloud computing and discusses possible solutions to these problems.

Leena Khanna and Anant Jaiswal [4] provided a brief overview and comparison of Cryptographic algorithms, with an emphasis on Symmetric algorithms which should be used for Cloud based applications and services that require data and link encryption. In this paper we review Symmetric and Asymmetric algorithms with emphasis on Symmetric Algorithms for security consideration on which one should be used for Cloud based applications and services that require data and link encryption. Using the comparisons made in this paper the various encryption algorithms can be evaluated.

Siwan Tang, Linfang Zou and Xuehua Liao [5] describe how online judge could be used as a real-time grading system for programming contests in Chinese Universities. Various advantages and disadvantages of online judge have been discussed. Adrian Kosowski Michał Małafiejski and Tomasz Noiński [6] implementation of an online judge with security concerns has been studied and experience for four years of its use has been presented. Jianhua Wu, Shuangping Chen and Rongrong Yang,[7] discusses the problems involved in experimental teaching of programming and algorithmic design have been discussed and design, implementation of an online judge system as a solution to this problem have been discussed. Using the information on the various cloud computing concerns pointed out by Ramgovind S et al. [8] such as regulatory compliance and privileged access the appropriate cloud delivery method so as to increase awareness for these concerns in the cloud environment. The proposed approach for integrating security aspects such as security data centers and virtual clusters in the cloud environment using reputation based systems by Kai Hwang et al. [10] allows safeguarding data objects at the file access level. How an advanced security model should not compromise the functionality or capabilities of in the present model is discussed by S. Subashini, V. Kavitha [11]. A new environment can also pose risks of data breaches which should be attentively understood by cloud service users.

III. Security Issues

The Online Judge may also have various security issues and they are mentioned below:

3.1 Authorization

Since, the Online Judge is meant to be used in institutions for assignment grading and contest hosting, proper authorization process needs to be incorporated. Users must be provided rights carefully in the Online Judge. Each user must be provided a separate identification so as to avoid any attempted breaches in each user's personal database. Using these unique identifications each user can access his/her archived responses and the website environment for further use. For example, any participant may not be able to change the input/output or the problem statement and on the other hand, problem setter should not be able to participate in an ongoing contest.

3.2 Solution Access

The users/participants in an ongoing contest/assignments should not be able to view the solutions of other users as it may result in increased cheating cases and plagiarism. The tester/evaluator should have access to all of the codes.

3.3 Malicious Codes

The users may run malicious codes on the judge such as infinite loops hence consuming a lot of memory. The restrictions on the resources such as memory or time may result in getting away from these cases. There may also be cases of users spawning a large number of child processes and resulting in damaging the resources of the server. The use of malicious program code and terminal commands could compromise the online judge environment. To prevent such actions of any user the online judge is secured by the use of *Sandbox Libraries*. The sandbox libraries are an open-source suite of software components for C/C++ and Python developers to create automated profiling tools and watchdog programs. Some features of sandbox libraries are:

- Capture arguments and system calls invoked by programs in runtime, and block malicious actions through user-defined policy modules
- specify quota limit of resources allocated to the program
- minimize privileges of programs, and isolate their execution from critical parts of the operating system

3.4 Users may have access to the codes of other users during ongoing contest/assignments.

The users of the online judge system are strictly forbidden to access the program codes of other users for any contest or assignment as it would be considered plagiarism for any user to use other users' program code as their own for submission to the respective query on the online judge. A solution to this problem is further discussed.

To secure the program code of each user a cascaded encryption will be applied to each of the code. The 3DES (Triple Data Encryption Standard) encryption technique will be used for encrypting the program code of the users. The use of 3DES encryption implements data security over the network in case of any network threats or unauthorized access of the program code.

3.5 Network Attacks

Another security issue in Online Judge is the possibility of network attacks such as denial of service and others. Hacking is another issue in such systems. Automatic grading can be incorporated to remove all the problems mentioned above. In this paper we have revisited the implementation of online judge to provide a more secure and efficient Online Judge. The security risk of such environments is the possibility of making outgoing network connections that can be used for spamming the ip address which would lead to the host server forfeiting support to the online judge's ip. Such attacks are handled by the use of *iptables* that can block locally originating packets based on the UID of the process that created those packets.

IV. Implementation details

An Online judge at the core is a simple PHP or Python script which can test source codes in different programming languages and can give the following outputs: Accepted (AC), Wrong Answer (WA), Time Limit Exceeded (TLE), and Runtime Error. One way to implement an online judge is to use a 3rd party API. In this case, one just needs to build a website or application which calls the API at the time of compilation and execution of the program.

The other way is to write an API for the online judge. The basic tasks which this API will handle are:-

1. Take the source code and check for compilation error.
2. Execute the program using some hidden test files and collecting the output.
3. Check for runtime errors.
4. Test correctness of the code by matching the output with the hidden test cases.

For writing the API, we can use any scripting language which can call OS methods within the script. For example, PHP uses 'exec' statement to execute OS processes.

Some of the advanced tasks which the API needs to take care of are:-

1. Checking for TLE verdict. If the process is not completed within the given time limit for the particular problem, then it should stop executing the program at that point and give TLE verdict.
2. Checking for source limit. The size of the program must satisfy some constraint as each source file utilizes resources on the server.

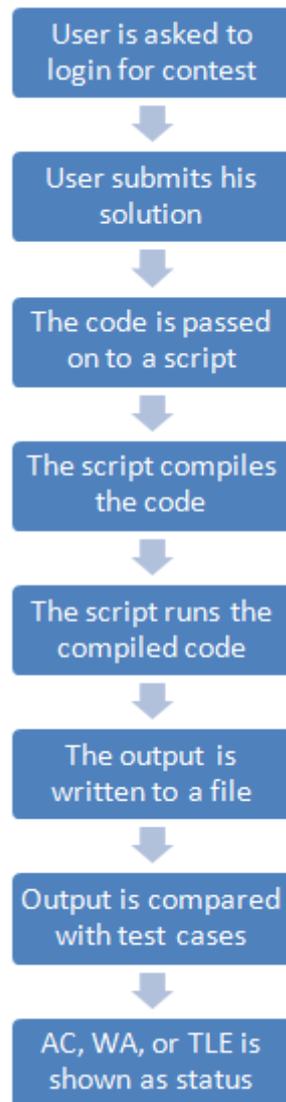


Fig. 2 Steps in the implemented judge

4.1 Online Part

To enable anyone on the Web to use the Online Judge, it has to be hosted on a public server. A private server can be used if the target users are of a single institution. A web application can be built using HTML, CSS, JavaScript, Bootstrap and some front-end frameworks. A database is needed for storing the problems, their test cases, users' data, etc.

The most difficult part of setting an Online Judge is to implement security at various levels. Some of the problems which can arise if security is not taken care of:-

1. Users can have full access to the servers and they can execute malicious processes on the systems.
2. Users can access the filesystem present on the servers.

One simple solution of protecting the servers from such attacks is to sandbox the executing code in a restricted environment so that a limited number of processes can be executed by the users.

4.2 Encryption Implementation

To secure the program code of each participant in the Online Judge system a 3DES encryption is applied to the program codes which is a cascaded block encryption technique. The 3DES encryption provides a 3 layered encryption using 3 distinct keys for each layer of the encryption. The cryptographic hash function used for the 3DES encryption in the Online Judge is the Secure Hash Algorithm 1 (SHA-1). This hash function provides a 168-bit (~21 bytes) hash value known as the message digest for the encryption. The cipher text production algorithm, using the program code as the plain text, is given by the following equation:

$$\text{Cipher Text} = \text{EK3} (\text{DK2} (\text{EK1} (\text{Plain Text})))$$

Where EK1 denotes encryption using key 1, DK2 denotes decryption using key 2, and EK3 denotes encryption using key 3 of the 3DES encryption.

Similarly for decrypting the encrypted program code on the secure end of the database on the Online Judge the following equation is executed:

$$\text{Plain Text} = \text{DK1} (\text{EK2} (\text{DK3} (\text{Cipher Text})))$$

Where DK3 denotes decryption using key 3, EK 2 denotes encryption using key 2, and DK 1 denotes decryption using key 1 which basically is the reverse of the pattern followed for encrypting the program code. Although SHA-1 takes more time compared to previous hashing functions due to the fact that it produces a 168-bit hash value, it provides a more secure encryption as it is not known to be broken and still believed to be secure.

V. Conclusion

An online judge has been successfully created which not only overcomes the disadvantages of a manual programming judge such as subjective nature, inefficient programming etc., but also takes care of the various security issues mentioned above. The online judge is able to successfully evaluate the code entered by the user and accordingly display results such as TLE (Time Limit Exceeded), CE (Compilation Error), WA (Wrong Answer), CA (Correct Answer) etc. The solutions or codes of each user are kept private and no one else but the user himself can access it. The online judge that has been implemented accepts code in multiple programming languages like C, C++, and Java etc.

Though using tools like online judge promote optimal programming, it may also lead to frustration among programmers as they may not be able to get all the test cases right for some cases. Such issues could be overcome by providing a bunch of old problems and blogs for programmers to get used to and learn more optimal style of programming. As we eliminate the human element in grading using this tool, the results are more precise and promote more effective solutions to problems.

References

- [1]. Andy Kurnia, Andrew Lim, Brenda Cheang, Online Judge, Published in Journal Computers & Education, Volume 36 Issue 4, Pages 299-315, 2001 (ISSN: 0360-1315)
- [2]. Akashdeep Bhardwaja, Subrahmanyam, Vinay Avasthi, Hanumat Sastry, Security Algorithms for Cloud Computing, Published in International Conference on Computational Modeling and Security Volume 85, Pages 535-542, 2016
- [3]. Wayne Jansen, Timothy Grance, Guidelines on Security and Privacy in Public Cloud Computing, Published in Special Publication (NIST SP) - 800-144, 2011
- [4]. Leena Khanna, Anant Jaiswal, "Cloud Computing: Security Issues and Description of Encryption Based Algorithms to Overcome Them", Published in IJARCSSE Volume 3, Issue 3, 2013 (ISSN:2277 128X)
- [5]. Siwan Tang, Linfang Zou, Xuehua Liao, "A Research on Online Judge Technology Based on MOOC Platform", Published in International Conference on Information Engineering and Communications Technology (IECT 2016) ISBN: 978-1-60595-375-5
- [6]. Adrian Kosowski, Michał Małafiejski, Tomasz Noiński, "Application of an Online Judge and Contester System in Academic Tuition", Published in Lecture Notes in Computer Science, vol 4823, 2008
- [7]. Jianhua Wu, Shuangping Chen, Rongrong Yang, "Development and application of online judge system", Information Technology in Medicine and Education (ITME), 2012
- [8]. Ramgovind S, Eloff MM, Smith E, The Management of Security in Cloud Computing, Published by Information Security for South Africa (ISSA), 2010 (ISBN: 978-1-4244-5493-8)
- [9]. Peter Mell, Timothy Grance, The NIST Definition of Cloud Computing, Published in NIST Special Publication 800-145, 2011
- [10]. S. Subashini, V. Kavitha, A survey on security issues in service delivery models of cloud computing, Published in Journal of Network and Computer Applications, Volume 34, Issue 1, Pages 1–11 (2011)
- [11]. Kai Hwang, Sameer Kulkarni, Yue HuCloud, Security with Virtualized Defense and Reputation-Based Trust Management, Published in Dependable, Published in Autonomic and Secure Computing, ISBN: 978-1-4244-5420-4 (2010)