IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Enhancing E-Learning System Performance And Availability Through Multiple Virtual Server Architecture: *A Case Study On Moodle Using Proxmox Virtualization And Reverse Proxy Techniques*

[1]Sujeet S. More , [2]Dr.Santosh S. Lomte

[1] Research Scholar, Dr. Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar, India ,
[2]Principal, Radhai Mahavidyalaya, Dr. Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar, India

Abstract: Currently, nearly all educational institutions rely on e-learning systems to support their learning processes. Performance is a critical factor in ensuring a smooth and comfortable user experience. While single-server architecture is commonly used to build these systems, it often proves inefficient due to its limited focus on availability. This study proposes a server architecture utilizing multiple virtual servers with Proxmox virtualization, employing reverse proxy techniques and storage clustering to enhance system availability for an e-learning Moodle platform. The design of the multiple virtual server architecture follows a methodology that includes preparation, planning, design, implementation, testing, and optimization. Research conducted on a testbed reveals that the multiple virtual server architecture offers superior availability compared to the single server setup. According to User Behavior Modeling Performance (UBMP) test results, the multiple virtual server architecture can handle up to 100 concurrent users with an availability level of 80.25%, whereas the single server architecture supports only 80 users with a lower availability level of 78.4%.

Index Terms **- Moodle, Reverse Proxy,MultiServer, Cluster.**

I. INTRODUCTION

An educational institution may operate multiple e-learning systems, typically utilizing a single-server architecture [1]. While a single-server setup is adequate for running one system, it poses risks when multiple systems are active simultaneously. If the server encounters an error, all services will cease, resulting in low system availability[2]. Additionally, performance and resource efficiency—both in terms of software and hardware—are critical factors that affect system performance and availability[3].

To address the limitations of single-server architecture, a multiple virtual server architecture is employed, coupled with reverse proxy techniques to enhance efficiency on a Proxmox server[4][5]. The reverse proxy allows multiple e-learning systems or applications to share a domain name and SSL certificate. It also provides load balancing and failover features, which significantly improve server availability[6]. Virtualization, a fundamental component of cloud computing under the Infrastructure as a Service (IaaS)[7][8] model, is also used to reduce hardware costs by optimizing hardware usage and minimizing electricity consumption, thus ensuring cost-effectiveness[9]. Proxmox Virtual Environment (PVE) is a high-availability hypervisor that manages virtual machines and can be used to create practical, web-based e-learning environments with noVNC technology[10][11]. Additionally, storage clustering techniques can be used to enhance scalability and availability by distributing storage across multiple redundant devices[12][13].

The single-server architecture has several limitations[14], as previously mentioned. In contrast, a multiple virtual server architecture divides a physical server into several virtual servers, optimizing hardware use and improving system availability by enhancing interactions between virtual servers[15]. The reverse proxy serves multiple purposes, such as cost efficiency and boosting system availability through load balancing, while storage clustering improves hardware availability[16][17]. Moodle is an effective and user-friendly platform for building e-learning systems, with proven performance benchmarks that minimize the risk of software bottlenecks[18].

In this research, Moodle is deployed as an e-learning application within a multiple virtual server architecture on Proxmox, with storage clustering implemented using GlusterFS[19]. The use of HAProxy as a reverse proxy aims to enhance resource efficiency and system availability on the server[20]. To validate the advantages of this architecture, a performance analysis is conducted using the User Behavior Modeling Performance (UBMP) method, which is comprehensive and adaptable to various server architectures, particularly in cloud environments[21].

The goal of this study is to apply the UBMP method to obtain effective, comprehensive, and accurate performance assessments of the tested architecture. A single-server architecture is also implemented for comparison with the multiple virtual server architecture to analyze performance and determine the reliability of each[22]. The test environment is implemented on a Proxmox server for each architecture using the Debian 8 operating system.

II. THEORETICAL REFERENCE

II.1 MOODLE

Moodle is an e-learning stage outlined to offer teachers, understudies, and chairmen a bound together framework for overseeing and conveying online instruction. E-learning epitomizes the concept of coordinate, [23]web-based remove learning that empowers free study[24], open through the web. Moodle makes computerized classrooms where clients can get to instructive materials and assets anytime, anywhere[25]. One of Moodleâs key focal points is its open-source nature, permitting clients with programming skill to customize and grow its highlights to meet particular needs and preferences[26].

II.2 REVERSE PROXY

The turn around intermediary strategy includes putting a intermediary server as an mediator between the client (frontend layer) and the server (backend layer). This procedure is commonly utilized for stack adjusting, dispersing the server stack successfully, and running caching highlights to decrease the require for rehashed substance conveyance, in this manner diminishing the internet server's workload[27]. Stack adjusting through a invert intermediary makes a difference minimize downtime, coming about in a system with high accessibility. HA Proxy could be a strong program arrangement for executing switch intermediary and stack adjusting, advertising dependable execution and flexibility in different utilize cases. These incorporate moderating space title allotments through space sending and executing SSL end, where scrambled communications are taken care of at the frontend layer, lessening the require for numerous SSL certificates[28].

II.3 VIRTUALIZATION

Virtualization could be an innovation that permits equipment assets to be made, overseen, and apportioned as required through program, empowering the establishment of different working frameworks on a single physical device[29]. The assets utilized in virtualization are determined from the physical computer, or have,

where the virtual machines are made. This asset allotment is made conceivable by hypervisor innovation, which oversees, runs, and screens virtual machines[30].

Figure 1: Hypervisor Proxmox Architecture.

Virtualization serves as the establishment for Foundation as a Benefit (IaaS) advertised by cloud computing providers[31]. Numerous cloud benefit suppliers, counting Amazon, utilize virtualization innovation to convey their administrations. The essential objective of virtualization is to optimize hardware usage and diminish power utilization. Proxmox could be a hypervisor known for its tall accessibility capabilities, competent of overseeing virtual machines with KVM and holders with LXC[32][33]. The engineering of Proxmox is outlined in Figure 1

II.4 TOOL FOR TESTING

There are right now a number of open-source apparatuses that can be utilized for push testing web servers or web administrations. In this article, we center on instruments reasonable for creating HTTP and HTTPS loads (GET, POST, and PUT) and FTP loads (PASV and RETR record exchanges). As it were apparatuses with conveyance open licenses (GNU, Apache Permit, etc.) and executables on the Linux bit working framework were chosen for execution investigation utilizing Apache JMeter [34]. Apache JMeter is open-source program utilized to perform stack testing, execution testing, and utilitarian testing on web applications and other web administrations [35].

II.5 USER BEHAVIOR MODELING PERFORMANCE (UBMP)

The scope of execution testing is broad, and varieties in design and assets altogether impact the angles being assessed. UBMP testing yields an execution show that's productive, comprehensive, and flexible, making it appropriate to a assortment of server models, especially in cloud situations. Key measurements for evaluating framework execution incorporate the ideal esteem and greatest concurrent clients. Conventional execution testing frequently faces challenges due to tall costs and constrained assets, making it troublesome to reach the greatest concurrent clients, and hence ruining the capacity to precisely gauge the system's crest stack capacity.

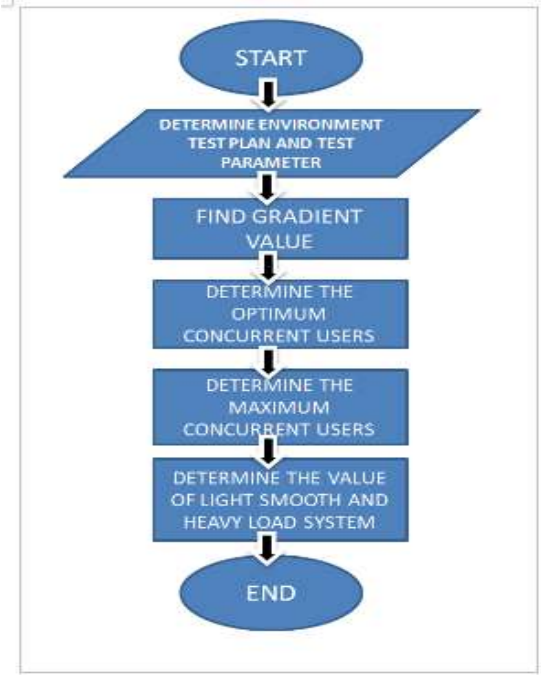

Figure 2: UBMP flowchart.

UBMP addresses these challenges by utilizing the intonation point execution file, which analyzes framework execution based on three markers: CPU utilization, throughput, and accessibility. UBMP categorizes framework execution into three portions based on the ideal esteem and greatest concurrent clients: light stack, smooth stack, and overwhelming stack. As the number of users increments, there's a comparing rise in CPU utilization and throughput until a certain limit, after which the accessibility level starts to decay. The ideal concurrent client esteem is recognized at the point where throughput is at its crest, whereas the greatest concurrent client esteem is decided when the system's accessibility drops to 70%.

Light stack conditions happen when the client check is underneath the ideal esteem, characterized by tall accessibility and progressive increments in throughput and CPU utilization. Smooth stack conditions exist inside the run between the ideal esteem and greatest concurrent clients. Overwhelming stack conditions emerge when the number of concurrent clients surpasses the greatest esteem.

Testing with UBMP joins different client behavior parameters to recreate practical conditions, such as considering time, ramp-up period, and a utilitarian test arrange. The UBMP testing handle to decide ideal values and most extreme concurrent clients, and to classify light, smooth, and overwhelming stack conditions, is portrayed in Figure 2.

STORAGE CLUSTERING

Capacity clustering may be a method that combines different capacity media into a coherent capacity unit, disseminating information over the clustered capacity media. This approach upgrades the scalability and accessibility of information capacity. Within the occasion of a capacity medium disappointment, the information remains secure since it is reproduced over other capacity media inside the cluster. One of the advances utilized for high-availability capacity replication is GlusterFS. In this consider, GlusterFS is connected to store client information for Moodle applications over two imitated web servers. The capacity clustering plot utilized is outlined in Figure 3.

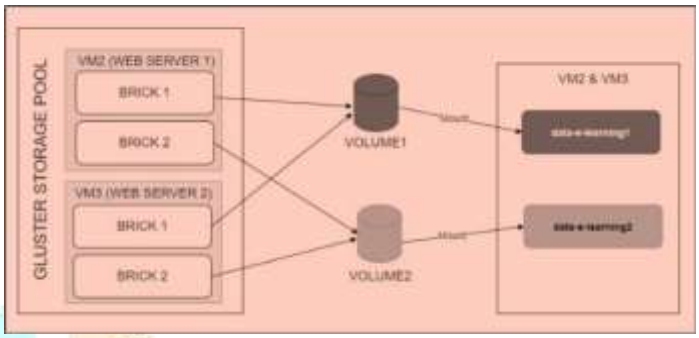

Figure 3: Storage Clustering Architecture.

III. MATERIALS AND METHODS

The strategy utilized in this consider was carried out within the taking after stages: planning, arranging, planning, actualizing, testing, and optimizing. This strategy comes about in a standard cycle of organize improvement and administration. The stream of the strategy is appeared in Figure 4.

The research stages are as follows:

1. Preparation
   This organize includes conducting a needs examination for cloud framework administrations and dissemination models, execution test models for cloud frameworks, Proxmox Virtual Environment frameworks, e-learning Moodle frameworks, and JMeter testing.
2. Planning
   Distinguish the organize engineering plans for both Single Server-Side and Different Virtual Server-Side Engineering frameworks.
3. Designing
   Create a arrange engineering topology, counting an e-learning Moodle framework, four virtualized servers, capacity clustering, a DNS server, and a web server, based on the testbed details.
4. Implementation
   Execute the organize engineering plan, covering establishment, organize tending to, and benefit setup, until the framework is completely arranged for utilize concurring to the testbed parameters.
5. Testing
   Test the built organize engineering, counting support, blame discovery, and execution observing utilizing testing instruments, taking after the parameters laid out within the test arrange.
6. Optimization
   Address any organize issues, optimize execution, and guarantee the organize adjusts with client needs and trade forms as indicated within the test arrange parameters.

III.1 TESTBED

Numerous virtual server-side models were outlined utilizing virtualization innovation in PVE to construct two particular e-learning frameworks fueled by Moodle. By actualizing a turn around intermediary strategy, both e-learning frameworks can be gotten to employing a single space title and SSL certificate, with the URLs https://lms.edu/lms1 for the primary e-learning stage and https://lms.edu/lms2 for the moment. The setup incorporates four virtualized servers on a physical machine: VM1 as the DNS and intermediary server, VM2 as Web Server 1, VM3 as Web Server 2, and VM4 as the database server. VM3 could be a copy of VM2, giving stack adjusting and tall accessibility. Capacity clustering is connected to the capacity media on VM2

Server	Processor	RAM	Storage(HDD)
Physical server (Proxmox)	AMD FX (tm)-6350 Six-Core	8GB	1TB
VM1 (DNS & Proxy)	1 Core	1 GB	8 GB
VM2 (Web Server 1)	2 Core	2 GB	8 GB
VM3 (Web Server 2)	2 Core	2 GB	8 GB
VM4 (Database Server)	1 Core	1 GB	8 GB

and VM3. All virtual servers run the 64-bit Debian 10 (Buster) working system with a command-line interface. The DNS server employments bind9, the intermediary server employments Haproxy, the net server employments NGINX, and the database server employments MariaDB. The plan of the numerous virtual server-side engineering testbed is outlined.

Table 1: Multiple Virtual ServerSide Architecture Specification.

The same case study in the form of deploying two e-learning applications and the same services on multiple virtual server-side architectures built in a testbed in the form of a single server-side architecture that uses specifications in the form of an accumulation of specifications used by virtual servers on multiple virtual server- side architectures as a comparison to analyze its performance. The testbed design on a single server-side architecture can be seen.

Hardware Specification			
CPU	RAM	Storage (Hard disk)	Network Interface Card
6 Core	8 GB	32 GB	Gigabit Ethernet

Table 2: Single Server Architecture Specification.

III.2 TEST PLAN

The test utilizes a stack testing method based on the UBMP show. Concurring to the UBMP flowchart in Figure 3, the primary step includes characterizing test plans and parameters. The test arrange incorporates getting to the begin page, logging in, and exploring to a course inside the e-learning framework. Tests were conducted at the same time on two e-learning courses, open at https://lms.edu/lms1 and https://lms.edu/lms2. The starting test parameters utilized to recognize the angle esteem are point by point in Table 3.

The slope esteem is decided by incrementally including 5 strings in each test cycle until a 5% increment in CPU utilization is watched. This slope esteem, speaking to the distinction in thread check between the primary

and nth emphasis that triggers a 5% CPU increment, is at that point utilized as a steady for including strings in consequent cycles to recognize the ideal and most extreme concurrent clients.

No	Parameter	Value
1	Concurrent users (Threads)	10
2	Ramp-up period	2
3	Loop	1
4	HTTP request protocol HTTPS	Threads / 1s
5	HTTP request host	lms.edu
6	HTTP request host path	/lms1, /lms2
7	Thinking time	1s - 9s (random)

Table 3: Test Parameters.

To discover the ideal number of concurrent clients, strings are included in each cycle, with consideration to throughput. The strings within the emphasis where most extreme throughput is accomplished are distinguished as the ideal concurrent clients. To decide the most extreme concurrent clients, testing starts with the ideal concurrent client check, at that point increments by the angle esteem in consequent emphasess until a framework accessibility level of up to 70% is come to. The intonation point execution record determined from this prepare makes a difference classify the system's stack conditionsâlight, smooth, or heavyâas an precise and productive execution metric.

IV. RESULTS AND DISCUSSIONS

The usage of a web-based online research facility has been conducted utilizing the Organized Control Framework Lab (NCSLab) system. In our investigate, the arrange engineering is sent in a cloud framework utilizing the Proxmox Virtual Environment (PVE). Tests were conducted based on two testbeds: the Different Virtual Server-Side Engineering and the Single Server-Side Design. A test arrange was at that point executed concurring to the parameters laid out in Table 3. The testing handle, taking after the Accessibility Testing stream for the Different Virtual Server-Side Design, is outlined in Figure 7. This test illustrates that the utilize of switch intermediary and capacity clustering not as it were diminishes framework working costs but moreover upgrades the accessibility of the framework and information [40].
Based on the testbed and test arrange for the framework, the slope esteem gotten is 10 concurrent clients, inferred from the distinction in strings between the 1st and 3rd emphasess, with a 5% distinction in CPU utilization, as appeared in Table 4 ("CPU Utilization in Angle Esteem Look"). The testing comes about demonstrate that server virtualization with Proxmox Virtual Environment (PVE) can viably optimize equipment execution on physical servers by partitioning them into different virtual servers, driving to tall productivity in asset utilization. The Proxmox server illustrated solidness amid these tests.

The test comes about too appear that both server models can ideally handle 50 clients, with a most extreme throughput of 4.6 seconds on numerous virtual servers and 4.9 seconds on a single server, as point by point in Table 5. The ideal comes about for concurrent clients are outlined in Figure 8, affirming that the concurrent client esteem is fitting [42].

Iterations	Threads	Multiple Virtual Server		Single Server	
		Availability	Error	Availability	Error
1	50	93.00%	7.00%	100%	0.00%
2	60	88.70%	11.30%	95.80%	4.20%
3	70	83.60%	16.40%	92.90%	7.10%
4	80	83.80%	16.20%	78.40%	21.60%
5	90	77.50%	22.50%	68.90%	31.10%
6	100	80.30%	19.70%	67.50%	32.50%
7	110	66.90%	33.10%	64.20%	35.80%
8	120	66.80%	33.20%	62.80%	37.20%

Table 4: Finding Optimum Concurrent Users Value Through Maximum Throughput.

The test comes about demonstrate that the numerous virtual server-side engineering beats the single server design in taking care of concurrent clients. The different server design can bolster up to 100 clients with an accessibility level of 80%, while the single server engineering oversees as it were 80 clients with an accessibility level of 78.4%, as appeared in Table 6. The ideal comes about for concurrent clients are outlined in

Iterations	Threads	Multiple Virtual Server		Single Server	
		Availability	Error	Availability	Error
1	10	2.5/s	0.00%	2.4/s	0.00%
2	20	3.3/s	0.00%	3.3/s	0.00%
3	30	4.1/s	0.00%	4.2/s	0.00%
4	40	4.1/s	1.50%	4.7/s	0.00%
5	50	4.6/s	7.00%	4.9/s	0.00%
6	60	4.5/s	11.40%	4.8/s	4.20%
7	70	4.3/s	16.40%	4.5/s	7.10%
8	80	4.0/s	16.20%	4.7/s	21.60%

Table 5: Finding Maximum Concurrent Users Value Through The Availability.

B ased on the comes about gotten from the inquire about, the execution of the numerous virtual server engineering is predominant to the single server design by dealing with 20 more clients within the framework within the test scope when the framework is tried for execution.

V. CONCLUSIONS

Based on inquire about information, the UBMP execution test show may be a comprehensive, compelling, and exact strategy for evaluating frameworks with web-based applications actualized within the Proxmox Virtual Environment (PVE). The integration of numerous virtual server-side structures, combined with switch intermediary and capacity clustering strategies, ideally utilizes existing assets, coming about in more prominent effectiveness, viability, and predominant execution compared to single-server designs. Whereas different virtual server-side designs are especially suited for e-learning frameworks, they can moreover be connected to other web-based applications. Assist improvement of the arrange engineering plan utilized in this research is essential to attain indeed more prominent execution and productivity within the usage of PVE..

VI. REFERENCES

[1] S.P.Chung, Y.J.Lu, and Y.C.Lai, “Cloud computing with single server threshold and double congestion thresholds,” ICT Express, vol.4,no.3, pp. 119– 123,2018,doi: 10.1016/j.icte.2017.03.002.
[2] I.Alharkan, M.Saleh, M.A.Ghaleb, H.Kaid, A.Farhan, and A.Almarfadi, “Tabu search and particle swarm optimization algorithms for two identical parallel machines scheduling problem with a single server,” J.King Saud Univ. - Eng.Sci.,vol.32,no. 5, pp. 330–338, 2020, doi: 10.1016/j.jksues.2019.03.006.
[3] P.Jain,Y.Munjal,J.Gera,and P.Gupta,“Performance Analysis of Various Server Hosting Techniques,” Procedia Comput.Sci.,vol.173,no. 2019, pp. 70–77, 2020, doi: 10.1016/j.procs.2020.06.010.
[4] R.Entezari-Maleki,L.Sousa, and A.Movaghar,“Performance and power modeling and evaluation of virtualizedserversinIaaSclouds,”Inf.Sci.(Ny).,vol.394–395,pp.106–122,2017,doi:10.1016/j.ins.2017.02.024.
[5] E.Ali, Susandri,and Rahmaddeni,“Optimizing Server Resource by Using Virtualization Technology,” Procedia Comput. Sci.,vol.59,no.Iccsci,pp.320–325,2015,doi: 10.1016/j.procs.2015.07.572.
[6] Z. Lei,H.Zhou,S.Ye,W.Hu,and G.P.Liu, “Cost-effective server-side re- deployment for web-based online laboratories using nginx reverse proxy,” IFAC- PapersOnLine, vol.53,no.2,pp.7204–17209, 2020, doi: 10.1016/j.ifacol.2020.12.1748.
[7] Y.Tao and G.Chen, “An extensible universal reverse proxy architecture,” Proc.- 2016 Int. Conf. Netw. Inf.Syst.Comput. ICNISC 2016, pp.8–11,2017,doi:0.1109/ICNISC.2016.45.
[8] A.Rashid and A.Chaturvedi, “Virtualization and its Role in Cloud Computing Environment,” Int. J.Comput. Sci.Eng.,vol.7,no.4,pp.1131–1136,2019,doi:10.26438/ijcse/v7i4.11311136.
[9] R.Kaur and S.Chopra, “Virtualization In Cloud Computing : A Review,” Int. J.Sci.Res.Comput. Sci.Eng.Inf.Technol., vol. 3307, pp. 01–05, 2020, doi:10.32628/cseit20641.
[10] V.Tiwari, A.A.Waoo, D.Akhilesh, A.Waoo,and B.Garg, “Study on Virtualization Technology and Its Importance in Cloud Computing Environment See Profile Study on Virtualization Technology and Its

Importance in Cloud Computing Environment," Int. J.Creat.Res.Thoughts,vol.8,no.2,pp.2320–2882,2020,[Online].Available: www.ijcrt.orgwww.ijcrt.org.
[11] M.K.L.Singh, W.A.T.Quijano, and G.Koneru, "Evaluation of network performance-type1 open source virtualization platforms," 2015Int.Conf.Comput.Commun.Informatics,ICCCI2015,2015, doi: 10.1109/ICCCI.2015.7218073.
[12] H.Mao,"High-density information security storage method of big data center based on fuzzy clustering,"Microprocess.Microsyst.,vol.81,no.November2020,p.103772,2021,doi: 10.1016/j.micpro.2020.103772.
[13] A.Pérez,G.Moltó,M.Caballer,and A.Calatrava,"Serverless computing for container-based architectures," Futur.Gener.Comput.Syst.,vol.83, pp.50–59,2018,doi:10.1016/j.future.2018.01.022.
[14] A.B.Prasetijo,E.D.Widianto, and E.T.Hidayatullah,"Performance comparisons of web server load balancing algorithms on HAProxy and Heartbeat," Proc. - 2016 3rd Int. Conf. Inf. Technol. Comput. Electr.Eng.ICITACEE 2016,pp.393–396,2017,doi:10.1109/ICITACEE.2016.7892478.
[15] J.E.C.De La Cruz and I.C.A.R.Goyzueta, "Design of a high availability system with HAProxy and domain name service for web services,"Proc.2017 IEEE 24th Int.Congr.Electron.Electr.Eng. Comput.INTERCON 2017,2017, doi:10.1109/INTERCON.2017.8079712.
[16] V.A.M and P.Sonpatki,ReactJS by Example:Building Modern Web Applications with React. 2016.
[17] L.L.Salekhova, K.S.Grigorieva, and T.A.Zinnurov, "Using LMS moodle in teaching CLIL:A case study,"Proc.-Int.Conf.Dev.eSystems Eng.DeSE,vol.October-20,no.May 2022,pp.393–395,2019,doi: 10.1109/DeSE.2019.00078.
[18] A.Villar-Martinez, L.Rodriguez-Gil,I.Angulo,P.Orduna, J.Garcia-Zubia,and D.Lopez-De-Ipina, "Improving the Scalability and Replicability of Embedded Systems Remote Laboratories through a Cost-EffectiveArchitecture,"IEEEAccess,vol.7,pp.164164–164185,2019,doi: 10.1109/ACCESS.2019.2952321.
[19] S.Kanthavar,S.Simeen,M.Chawla,and S.Sarayu,"Design of an Architecture for Cloud Storage to Provide Infrastructure as a Service (IaaS),"2017 14th IEEE India Counc.Int.Conf.INDICON 2017,2018, doi:10.1109/INDICON.2017.8488000.
[20] F.Zhang,Y.Liu,J.Mao,M.Zhang,and S.Ma,"User behavior modeling for Web search evaluation," AI Open, vol. 1, no. March, pp.40–56, 2020,doi:10.1016/j.aiopen.2021.02.003.
[21] H.H. Li,X.R.Li,H.Wang,J.Zhang, and Z.X.Jiang, "Research on cloud performance testing model," Proc.IEEE Int.Symp.High Assur.Syst.Eng.,vol.2019-Janua,no.January 2019,pp.179–183,2019, doi:10.1109/HASE.2019.00035.
[22] A.F.Wicaksono and A.Moffat,"Metrics, User Models, and Satisfaction," pp.654–662,2020.
[23] P.He,J.Qiu, and B.Zhai, "Study on the integration of cloud computing and moodle learning platform," Proc. 2015 IEEE Int. Conf. Commun. Softw. Networks, ICCSN 2015, pp.367–371,2015,doi: 10.1109/ICCSN.2015.7296185.
[24] C.DeMedio,C.Limongelli,F.Sciarrone,and M.Temperini,"MoodleREC:A recommendation system for creating courses using the moodle e-learning platform," Comput. Human Behav.,vol.104,p. 106168,2020,doi:10.1016/j.chb.2019.106168.
[25] M.Zabolotniaia, Z.Cheng,E.M.Dorozhkin, and A.I.Lyzhin,"Use of the LMS Moodle for an effective implementation of an innovative policy in higher educational institutions," Int.J.Emerg. Technol.Learn.,vol.15,no.13,pp.172– 189,2020,doi:10.3991/ijet.v15i13.14945.
[26] W.Fernando, "Moodle quizzes and their usability for formative assessment of academic writing," Assess.Writ.,vol.46,no.September, p.100485,2020,doi:10.1016/j.asw.2020.100485.
[27] D.Arnaldy and T.S.Hati, "Performance Analysis of Reverse Proxy and Web Application Firewall with Telegram Bot as Attack Notification on Web Server," 2020 3rd Int.Conf.Comput.Informatics Eng. IC2IE 2020,pp.455–459,2020,doi:10.1109/IC2IE50715.2020.9274592.
[28] L.H.Pramono,R.C.Buwono,and Y.G.Waskito, "Round-robin algorithm in HAProxy and nginx load balancing performance evaluation: A review," 2018 Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2018, no.July, pp.367–372,2018,doi:10.1109/ISRITI.2018.8864455.
[29] K.Kritikos and P.Skrzypek,"Simulation-as-a-service with serverless computing," Proc.-2019 IEEE World Congr.Serv.Serv.2019,vol.2642–939X,pp.200–205,2019,doi: 10.1109/SERVICES.2019.00056.
[30] L.Hernandez, G.Jimenez, A.Pranolo, and C.U.Rios,"Comparative Performance Analysis Between Software-Defined Networks and Conventional IP Networks," Proceeding - 2019 5th Int.Conf.Sci.Inf. Technol.Embrac.Ind.4.0 Towar.Innov.Cyber Phys.Syst.ICSITech2019,pp.235–240,2019,doi: 10.1109/ICSITech46713.2019.8987493.

[31] K.Arjunan and C.N.Modi,"An enhanced intrusion detection framework for securing network layer of cloud computing," ISEA Asia Secur.Priv.Conf.2017,ISEASP2017,2017,doi: 10.1109/ISEASP.2017.7976988.
[32] J.Yang and Y.Lan, "A performance evaluation model for virtual servers in KVM-based virtualized system," Proc. - 2015 IEEE Int. Conf. Smart City, SmartCity 2015, Held Jointly with 8th IEEE Int.Conf. Soc.Comput. Networking,Soc.2015,5th IEEE Int.Conf.Sustain.Comput.Communic, pp.66–71,2015, doi:10.1109/SmartCity.2015.49.
[33] Y.Ariyanto,B.Harijanto,A.Setiawan,S.Adhisuwignjo,and E.Rohadi, "Integration of digital library server with Service Oriented Architecture (SOA) based on cloud computing using proxmox server," J. Phys. Conf. Ser.,vol.1402, no.7, 2019, doi:10.1088/1742-6596/1402/7/077054.
[34] S.Grabovsky, P.Cika, V.Zeman,V.Clupek, M.Svehlak, and J.Klimes,"Denial of Service Attack Generator in Apache JMeter,"Int.Congr.Ultra Mod.Telecommun.Control Syst.Work.,vol.2018-November, pp.1–4, 2019, doi: 10.1109/ICUMT.2018.8631212.
[35] P.Cika, S.Grabovsky, V.Zeman, and V.Clupek, "Network Emulator of Transmission Parameters of Data Networks," Int. Congr. Ultra Mod. Telecommun. Control Syst. Work.,vol.2018-November, no. 1, pp.1–6,2019, doi:10.1109/ICUMT.2018.8631254.
[36] S.Pradeep and Y.K.Sharma, "A Pragmatic Evaluation of Stress and Performance Testing Technologies for Web Based Applications," Proc. - 2019 Amity Int.Conf.Artif.Intell.AICAI2019,pp. 399–403,2019,doi:10.1109/AICAI.2019.8701327.
[37] O.Hkdylru et al.,"Cloud Storage Behavior Analysis Using Time Series Clustering," pp.90–95, 2018.
[38] Z.Tang,S.Sha,J.Zhang,Z.Li,H.Peng, and H.Du,"WebGlusterFS2.0: A Web-based Administration Tool for GlusterFS with Unattended Deployment and Automatic Configuration of Xlator," 2019 Comput. Commun. IoT Appl.ComComAp2019,pp.276–281,2019,doi: 10.1109/ComComAp46287.2019.9018744.
[39] Q.Ye, W.Hu, and H.Zhou, "Implementation of WebVR-based laboratory for control engineering education based on NCSLab framework," Chinese Control Conf.CCC,pp.7880–7885, 2017,doi:10.23919/ChiCC.2017.8028601.