RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

ENHANCING CLOUD SERVICE RELIABILITY WITH LOW THROUGHPUT DATA TRANSMISSION

[1]Veta. Chaitanya,[2]Chatragadda. Prasanna, [3]thambala.Ramesh,[4]S.Sowmya Reddy
[123]Assistant Professor,[4]Student
Department of CSE Engineering
Abdul Kalam Institute of Technological Sciences, Kothagudem, Telangana

ABSTRACT

It is important to demonstrate a distributed computing community in order to evaluate and predict its internal availability, consistency, and usability. Many previous studies on the accessibility of virtualized systems and the evaluation of their consistent quality with respect to specific servers in cloud server farms have been reported. In this research, we present a different tiered demonstrating method for evaluating the accessibility and reliability of server farms that are tree-based. The progressive model consists of three layers: (i) an issue tree that illustrates the subsystems' engineering; (ii) unwavering quality charts in the top layer that display the framework network geography; and (iii) stochastic prize nets that comprehensively capture the behaviors and dependencies of the subsystems' components. In the context of three-level and fat-tree geographies, two agent server farm networks are presented and thoroughly examined. We include many contextual studies directly in order to investigate the impact of the board and systems management on distributed computing environments. Additionally, we do several detailed analyses related to consistent quality and accessibility metrics for the framework models. The analysis's findings demonstrate how appropriate systems management may increase dependability and accessibility by bettering the distribution of hubs within server farm organizations. The results of this research may be used to the formation and responsible management of distributed computing communities..

1. INTRODUCTION

In modern ICT ecosystems, data center (DC)s play the role of a centric core. The huge network system of physical servers in DCs (also known as the data center network (DCN)) facilitates the continuous operation of online businesses and information services from distant parts of the world. Under strict requirements to mitigate any catastrophic failures and system outages, DC systems are in the progress of rapid expansion and redesign for high reliability and availability. The reliability availability of a certain server system in DCs is commonly supposed to be dependent on the reliability/availability of its own physical subsystems as well as the number of subsystems involved in the system architecture. However, because every compute node in a DCN communicates with other nodes via a network topology, it is a matter of curiosity that different manipulations of a certain system with similar components can gain different

measures of interest. Thus, even though the number of components remains unchanged, their appropriate allocation and networking can significantly improve the reliability/availability of the system. Few studies on the extent to which the allocation and interconnection of subsystems can affect the reliability/availability of the overall system in DCNs have been published.

An appropriate architecture to interconnect the physical servers in a DCN is important for the agility and recongurability of DCs. The DCNs are required to respond to heterogeneous application demands and service requirements with high reliability/availability as well as high performance and throughput. Contemporary DCs employ top of rack (ToR) switches interconnected through end of rack (EoR) switches, which are, in turn, connected to core switches. Nevertheless, recent studies proposed a variety of network topology designs in which each approach features its unique network architecture, fault avoidance and recovery, and routing algorithms. We adopt the architecture classification of DCN presented in to categorize DCNs into three main classes: (i) switch-centric architectures, for instance, Three-tier, Fat-Tree, PortLand, and F2Tree ; (ii) server-centric architectures (also known as recursive topologies) e.g, DCell, Ficonn, MCube, and (iii) hybrid/enhanced architectures, e.g., Helios.

In practice, four main network topologies are widely used to construct server networks in DCs including two switch centric topologies (three-tier and fat-tree), and two server centric topologies (BCube, DCell). Among these topologies, fat-tree (and its variants) is a potential candidate of DCN topologies for mass-built DCs of giant online-business enterprises such as Google and Facebook. The use of a large number of small, commodity and identical switches help reduce the construction budget for a new DC significantly while balancing other measures and characteristics of a DCN. The small and identical switches differ only in their configuration and placement in the network, but they deliver low power bandwidth operational expenditure (OPEX) and capital expenditure (CAPEX). Furthermore, the deployment of pods in fat-tree topology can be incremental without any downtime or rewiring when the size of DC is requested to scale/built out. Also, network softwares are not required to be written to be network aware when considering a good performance, which is the biggest advantage of fat-tree topology. Cabling complexity is, however the daunting disadvantage of the fat-tree topology in practical deployment.

In comparison to other relevant DCN topologies, fat-tree outperforms in various measures. For instance, fat-tree is better than DCell and BCube in terms of some performance related metrics such as throughput and latency. In comparison with three-tier topology, fat-tree DCNs do not require the use of high-end switches and high-speed links, thus can drop the total deployment cost rapidly. In general, the common metrics to assess a DCN in practice are scalability, path diversity, throughput and latency, power consumption, and cost. More recently, to maintain long running

online services, the ability of DCNs to tolerate multiple failures (of links, switches and compute nodes) is an essential characteristic requiring urgent consideration for DCNs. Thus, appropriate modeling and evaluation of the fault-tolerance characteristics using stochastic models are necessary to enhance the reliability/availability for DCNs.

In this paper, we focus on exploring fault-tolerant indicators of connectivity in a DCN including reliability/availability for the simplest non-trivial instance of fat-tree topology (as a widely-used candidate in industry) in comparison with three-tier topology (contemporarily used in many giant DCs) using stochastic models. A failure of network elements in DCNs is inevitable. Therefore, the network requires automatic reconfiguration mechanisms and restoration of network services at the moment of failure until a complete repair of the faults of nodes/links becomes possible. Service outages due to any type of failures in a DC significantly incur huge costs on both providers and customers. A study carried out by Ponemon Institute among 63 DCs shows that, the average cost since 2010 due to downtime of each DC has increased 48% from 500,000USD to 740,357USD. In addition, according to a report on failure rates within the Google clusters of 1,800 physical servers (used as building blocks in the IT infrastructure of Google Data Centers), there are roughly 1,000 individual machine failures and thousands of hard drive failures in each cluster during the first year of operations, also the cost to repair each failure reaches almost 300USD, not considering the losses caused directly by the failure in terms of operational business revenues. Thus, reliability/availability evaluation of a cloud-based DC requires a comprehensive model in which different types of failures and factors causing the failures are necessarily taken into account. The detailed analysis of such models could also help technicians to choose appropriate routing policies in the deployment of IT infrastructure.

2. LITERATURE SURVEY

Cloud Computing

(A.Abbas and S. U. Khan, July 2014)

Cloud computing is emerging as a new computing paradigm in the healthcare sector besides other business domains. Large numbers of health organizations have started shifting the electronic health information to the cloud environment. Introducing the cloud services in the health sector not only facilitates the exchange of electronic medical records among the hospitals and clinics, but also enables the cloud to act as a medical record storage center. Moreover, shifting to the cloud environment relieves the healthcare organizations of the tedious tasks of infrastructure management and also minimizes development and maintenance costs. Nonetheless, storing the patient health data in the third-party servers also entails serious threats to data privacy. Because of probable disclosure of medical records stored and exchanged in the cloud, the patients' privacy concerns should essentially be considered when designing the security and privacy mechanisms. Various approaches have been used to preserve the

privacy of the health information in the cloud environment. This survey aims to encompass the state-of-the-art privacy-preserving approaches employed in the e-Health clouds. Moreover, the privacy-preserving approaches are classified into cryptographic and noncryptographic approaches and taxonomy of the approaches is also presented. Furthermore, the strengths and weaknesses of the presented approaches are reported and some open issues are highlighted.

(J. Pecarina, S. Pu and J. Liu, 2012)

Existing cloud storage systems lack privacy aware architectures that meet accessibility goals for complex collaboration. This deficiency is fully realized in the healthcare industry, where cloud-enabling technology blurs the ownership boundary of health and wellness information. Whether among traditional `stovepiped' data silos, health information exchanges or personally controlled health information repositories, various forms of privacy neglect are common practice. We propose a paradigm shift in the interaction of users with cloud services that removes unwarranted trust in the cloud service provider and provisions accessibility for collaborators. To realize the paradigm shift, it is necessary to provide anonymity in data storage and separate the administration of access policy and authorization from the mechanisms used for enforcement. The dispensation of authorizations in the SAPPHIRE architecture bootstraps a traditional Kerberos ticket-based approach with `trust verifications'. In our evaluation, we prove the security properties of the SAPPHIRE architecture and implement a small scale prototype. Our analysis shows that SAPPHIRE is a viable extension of collaborative health information systems through the provision of anonymity and enhanced policy administration for the primary data owner.

(A.N. Khan, M. L. M. Kiah, S. U. Khan, S. A. Madani and A. R. Khan, 2013)

While using the cloud storage services on resource constraint mobile device, the mobile user needs to ensure the confidentiality of the critical data before uploading on the cloud storage. The resource limitation of mobile devices restricts mobile users for executing complex security operations using computational power of mobile devices. To make security schemes suitable for mobile devices, large volume of existing security schemes execute complex security operations remotely on cloud or trusted third party. Alternatively, few of the existing security schemes focus on the reduction of the computational complexity of the cryptographic algorithms. Keeping in view the resource limitation of mobile devices, this paper, introduces an incremental cryptographic version of the existing security schemes, such as encryption-based scheme, coding-based scheme, and sharing-based scheme, for improving the block(s) modification operations in term of resource utilization on mobile device. The experimental results show significant improvement in resource utilization on mobile device while performing block insertion, deletion, and modification operations as compared to the

original version of the aforementioned schemes.

Reliability and availability of cloud computing

(R. C. Andes and W. B. Rouse, 1990)

Adaptive aiding, a concept that involves tailoring the time and nature of operator aid to variation of tasks, operators, and environments, is examined. Aiding possibilities are discussed from the perspective of application domains and the need to integrate adaptive aiding with other intelligent systems. Particular attention is given to the attributes affecting the specification process. ADAPT, a design tool for assisting designers in conceptualizing and specifying functionality of adaptive aiding systems, is described. Emphasis is placed on a proposed scenario analysis facility design and analysis of the specification process. ADAPT's shortcomings are briefly discussed

(P. A. Hancock and M. H. Chignell, 1988)

In examining the role of time in mental workload, the authors present a different perspective from which to view the problem of assessment. Mental workload is plotted in three dimensions, whose axes represent effective time for action, perceived distance from desired goal state, level of effort required to achieve the time-constrained goal. This representation allows the generation of isodynamic workload contours that incorporate the factors of operator skill and equifinality of effort. An adaptive interface for dynamic task reallocation is described that uses this form of assessment to reconcile the joint aims of stable operator.

(D. Bailey, E. Frank-Schultz, P. Lindeque and J. L. Temple, III, 2008)

We present a brief introduction to three reliability engineering techniques: failure mode, effects, and criticality analysis; reliability block diagrams; and fault tree analysis. We demonstrate the use of one of these techniques, reliability block diagrams, in evaluating the availability of information technology (IT) systems through a case study involving an IT system supported by a three-tier Web-server configuration.

(R. d. S. Matos, P. R. M. Maciel, F. Machida, D. S. Kim and K. S. Trivedi, 2012)

Server virtualization is a technology used in many enterprise systems to reduce operation and acquisition costs, and increase the availability of their critical services. Virtualized systems may be even more complex than traditional nonvirtualized systems; thus, the quantitative assessment of system availability is even more difficult. In this paper, we propose a sensitivity analysis approach to find the parameters that deserve more attention for improving the availability of systems. Our analysis is based on Markov reward models, and suggests that host failure rate is the most important parameter when the measure of interest is the system mean time to failure. For capacity oriented availability, the failure rate of applications was found to be

another major concern. The results of both analyses were cross-validated by varying each parameter in isolation, and checking the corresponding change in the measure of interest. A cost-based optimization method helps to highlight the parameter that should have higher priority in system enhancement.

3. IMPLEMENTATION AND RESULT ANALYSIS

The proposed system is implemented with the following modules. Data Owner In this module, the data owner uploads their data in the cloud server. For the security purpose the data owner store in the particular Sub Systems (SS1 and SS2) and base station will connect to neighbor nodes and then file will store in smallest distance neighbor node. After storing data owner will verify the file is safe or not. The Data owner can have capable of manipulating the data file.

Cloud Servers

The cloud server is responsible for data storage and file authorization for an end user. The data file will be stored in a particular base stations (SS1 and SS2) and neighbor nodes with their tags such as file name, secret key, digital sign, and owner name. If the end user requested file is correct then the data will be sent to the corresponding user and also will check the file name, end user name and secret key in all Base stations and neighbor nodes. If all are true then it will send to the corresponding user or he will be captured as attacker.

Data Center

DATA CENTER Server means Location Based Services. In DATA CENTER server Base stations (SS1 and SS2) and neighbor nodes are present. Data Center Server is a cloud which is responsible for handling the all Base stations (SS1 and SS2) and neighbor nodes. In Data Center server Data owner can view the files, attacker details, file search and response details, view node distance and Unblock user. The data file will be stored in DATA CENTER Server under particular base stations (SS1 and SS2) and neighbor nodes. The end user can request the file to DATA CENTER server and it will connect to particular base stations (SS1 and SS2) and neighbor nodes. If the requested file is found then send to end user. Data Consumer (End User) The data consumer is nothing but the end user who will request and gets file contents response from the corresponding cloud servers or DATA CENTER server. Before downloading any files from the server, end user has to request a secret key of particular file. If the file name and secret key is correct then the end user is getting the file response from the DATA CENTER server or else he will be considered as an attacker and also he will be blocked in corresponding **DATACENTER** server. If he wants to access the file after blocking he wants to UN block from the DATA CENTER server.

Attacker

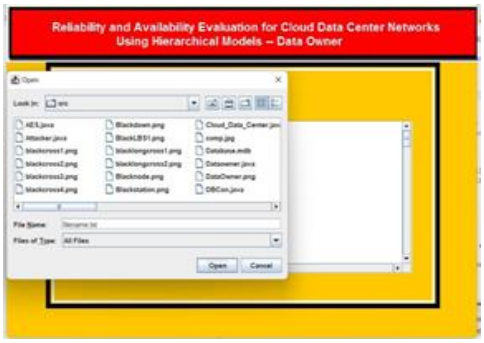
Attacker is one who is integrating the DATA CENTER server file by adding malicious data to the corresponding file. The may be within a DATA CENTER server or from outside the DATA CENTER server.

4. SCREEN SHOTS

This is the screen to browse a File.

Click "Browse" to get below screen for browsing a file.

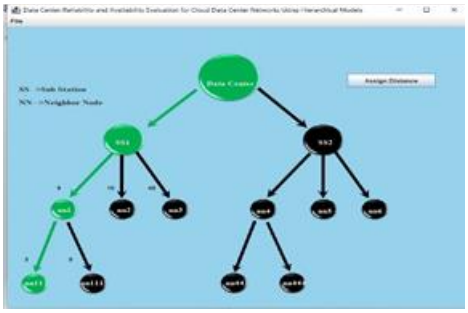
Select a file to upload from the above screen.



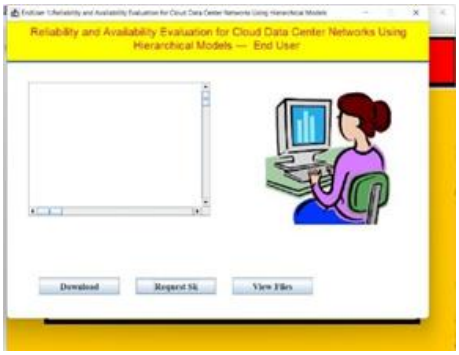
After selecting a file, file will be displayed as above.

To upload the file selected, Click "Upload". Select Base Station to upload the file.

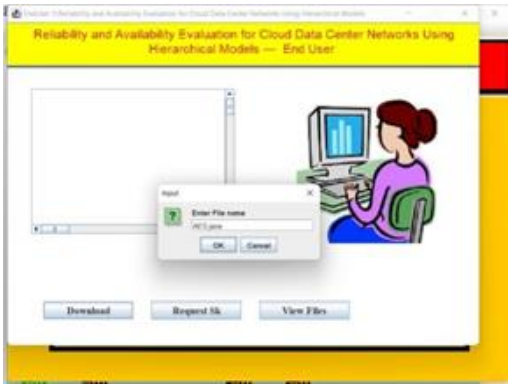
Path of file being uploaded in the base station selected.



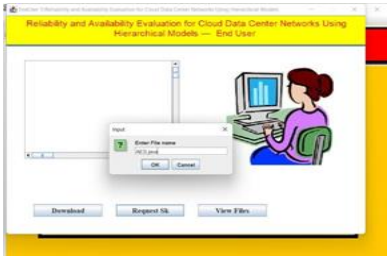
To request secret key generation to download a file, click "Request SK".



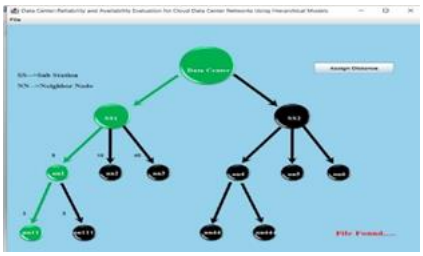
Enter the file name to get secret key.



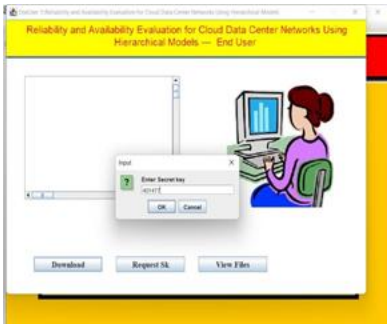
Secret key is generated for the above selected file as displayed in the above screen.



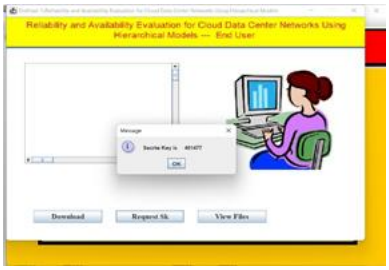
Enter the file name to download.



Enter the secret key generated in the previous steps.



If the secret key entered is correct then next below screen will be displayed.



After entering correct secret key, file is found as shown above.

5. CONCLUSION AND FUTURE ENHANCEMENT

A thorough hierarchical modeling and study of DCNs was reported in this article. The systems are built on tree-based switch-centric network topologies (fat-tree and three-tier), which include sixteen physical servers accompanied by three levels of switching switches. With an RG at the system layer, a fault-tree at the subsystem layer, and an SRN at the component layer, we tried to build hierarchical models for the system. In addition, we carried out many thorough assessments with respect to availability and dependability. The findings demonstrated that improving the network's active node distribution may improve cloud computing systems' availability and dependability. Moreover, the primary influencing elements are the MTTF and MTTR of the physical servers, while the connections play a crucial role in preserving the system's high availability. This study's findings may help with the design and administration of useful cloud computing facilities.

5. REFERENCES

[1] M. F. Bari et al., "Data center network virtualization: A survey," IEEE Commun. Surveys Tuts., vol. 15, no. 2, pp. 909_928, 2nd Quart., 2013. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper. htm?arnumber=6308765

[2] R. Cocchiara, H. Davis, and D. Kinnaird, "Data center topologies for mission-critical business systems," IBM Syst. J., vol. 47, no. 4, pp. 695_706, 2008. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5386510

[3] T. Chen, X. Gao, and G. Chen, "The features, hardware, and architectures of data center networks: A survey," J. Parallel Distrib. Comput., vol. 96, pp. 45_74, Oct. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0743731516300399

[4] S. Zafar, A. Bashir, and S. A. Chaudhry, "On implementation of DCTCP on three-tier and fat-tree data center network topologies," Springer- Plus, vol. 5, no. 1, p. 766, Dec. 2016. [Online]. Available: http://springerplus.springeropen.com/articles/10.1186/s40064-016-2454-4

[5] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," ACM SIGCOMM Comput. Com- mun. Rev., vol. 38, no. 4, pp. 63_74, 2008. [Online]. Available: http://dl.acm.org/citation.cfm?id=1402958.1402967

[6] R. N. Mysore et al., "PortLand: A scalable fault-tolerant layer 2 data center network fabric," in Proc. ACM SIGCOMM Conf. Data Com- mun. (SIGCOMM), 2009, pp. 39_50. [Online]. Available: http://doi.acm.org/10.1145/1592568.1592575

[7] G. Chen, Y. Zhao, D. Pei, and D. Li, "Rewiring 2 links is enough: Accelerating failure recovery in production data center networks," in Proc. 35th IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS), Jun. 2015, pp. 569_578. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7164942

[8] Y. Liu, D. Lin, J. Muppala, and M. Hamdi, "A study of fault-tolerance characteristics of data center networks," in Proc. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops (DSN), Jun. 2012, pp. 1_6. [Online]. Available: http://ieeexplore.ieee.org/document/6264696/

[9] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, "Dcell: A scalable and fault-tolerant network structure for data centers," in Proc. ACM SIGCOMM Conf. Data Commun. (SIGCOMM), vol. 38, no. 4, Aug. 2008, pp. 75_86. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1402958.1402968

[10] D. Li, "FiConn: Using backup port for server interconnection in data centers," in Proc. IEEE 28th Conf. Comput. Commun. (INFO- COM), Apr. 2009, pp. 2276_2285. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5062153

[11] C. Wang, C. Wang, Y. Yuan, and Y. Wei, "MCube: A high performance and fault-tolerant network architecture for data centers," in Proc. Int. Conf. Comput. Design Appl., Jun. 2010, pp. V5-423_V5-427. [Online]. Available: http://ieeexplore.ieee.org/document/5540940/

[12] N. Farrington et al., "Helios: A hybrid electrical/optical switch architecture for modular data centers," ACM SIGCOMM Comput. Com- mun. Rev., vol. 40, no. 4, pp. 339_350, Aug. 2010. [Online]. Available: http://dl.acm.org/citation.cfm?doid=1851275.1851223

[13] H. M. Helal and R. E. Ahmed, "Performance evaluation of datacenter network topologies with link failures," in Proc. 7th Int. Conf. Modeling, Simulation, Appl. Optim. (ICMSAO), Apr. 2017, pp. 1_5. [Online]. Available: http://ieeexplore.ieee.org/document/7934898/

[14] N. Farrington and A. Andreyev, "Facebook's data center network architecture," in Proc. IEEE Opt. Interconnects Conf., May 2013, pp. 49_50. [Online]. Available: http://ieeexplore.ieee.org/document/6552917

[15] B. Lebiednik, A. Mangal, and N. Tiwari. (May 2016). "A survey and evaluation of data center network topologies." [Online]. Available: http://arxiv.org/abs/1605.01701

[16] F. Yao, J. Wu, G. Venkataramani, and S. Subramaniam, "A comparative analysis of data center network architectures," in Proc. IEEE Int. Conf. Commun. (ICC), Jun. 2014, pp. 3106_3111. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6883798

[17] Ponemon Institute and Emerson Network Power. 2013 Cost of Data Center Outages. Accessed: Oct. 12, 2018. [Online]. Available: http://www.emersonnetworkpower.com/documentation/enus/brands/liebert/documents/whitepapers/2013_emerson_data_center_cost_downtime_sl-24680.pdf

[18] R. Miller. (2008). Failure Rates in Google Data Centers. Data Center Knowledge, Business. Accessed: Oct. 20, 2018. [Online]. Available: https://www.datacenterknowledge.com/archives/2008/05/30/failurerates-in-google-data-centers

[19] T. Lumpp et al., "From high availability and disaster recovery to business continuity solutions," IBM Syst. J., vol. 47, no. 4, pp. 605_619, 2008. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5386516

[20] D. M. Gomes et al., "Evaluating the cooling subsystem availability on a cloud data center," in Proc. IEEE Symp. Comput. Commun. (ISCC), Jul. 2017, pp. 736_741. [Online]. Available:

http://ieeexplore.ieee.org/document/8024615/