QUID 2017, pp. 460-474, Special Issue N°1- ISSN: 1692-343X, Medellín-Colombia

PROVIDING A NOVEL ALGORITHM IN SDDC'S STORING BASED ON FAULT TOLERANCE AND HIGH AVAILABILITY

(Recibido el 15-06-2017. Aprobado el 04-09-2017)

Mohammadreza Shams,
Department of Information Technology Engineering, E-Campus, Islamic Azad University,
Tehran, Iran
shams@iauec.ac.ir

Mohammadreza Majma
Department of Computer Engineering, Pardis Branch, Islamic Azad University,
Pardis, Iran
m_majma@pardisiau.ac.ir

RESUMEN: Las cuestiones como el aumento de la velocidad de transmisión, la reducción de los costos de transmisión de datos, la seguridad de los datos y el tráfico de red son importantes y considerables en la transmisión o replicación de datos entre dos centros de datos definidos por software; Estos centros pueden lograr una mayor fiabilidad y disponibilidad y garantizar servicios sin escalas. En este trabajo, hemos propuesto un algoritmo para dividir y cifrar los datos que es producido por el servicio (SDDC) proveedores de software definido centro de datos sobre la base del tipo de importancia. Transferimos datos más importantes con RSA y menos importantes con algoritmos DES con encriptación, y también datos de cada proveedor de servicios sin cifrado, a diferentes partes de almacenamiento, simultáneamente. Los resultados de los experimentos muestran que el algoritmo propuesto reduce el tráfico de red en un 67%, el volumen de datos cifrados en un 55% y el costo de adquisición de ancho de banda en un 50%. Sin embargo, en algunos casos, el cifrado de datos requiere mucho tiempo.
PALABRAS CLAVE: SDDC, SDS, algoritmos de cifrado, fiabilidad, integridad de datos
.

ABSTRACT: Issues like increasing the speed of transmission, reducing the costs of data transmission, data security and network traffic are important and considerable in transmission or replication of data between two software-defined data centers; these centers can achieve to higher reliability and availability and ensure non-stop services. In this paper, we proposed an algorithm to divide and encrypt the data which is produced by service (SDDC) providers of software-defined data center based on the type of importance. We transferred more important data with RSA and less important one with DES algorithms with encryption, and also data of each service providers without encryption, to different storing parts, simultaneously. The results of experiments show that the proposed algorithm reduces network traffic by 67%, the volume of encrypted data by 55% and the cost of bandwidth procurement by 50%. However, in some cases, data encryption is time-consuming.
KEYWORDS: SDDC, SDS, Encryption algorithms, reliability, Data integrity

1. INTRODUCTION

With the rise in demand for cloud services, massive amount of contents have been created and shared over the cloud networks (i.e., networks of data centers). Thus, ensuring disaster-resilient data-center (DC) networks has become a major requirement for network providers. Natural disasters such as earthquakes, tornadoes, etc., and human-made disasters such as weapons of mass destruction (WMD) attacks pose a major threat to DC networks since failures due to these attacks are correlated and cascading, and can cause huge amount of data loss and service disruptions [1] [2]. Increasing use of software-defined data centers is quite significant since they would reduce CAPEX (capital expenditure) up to 75% and OPEX (operational expenditure) up to 55% and finished costs in 3 years up to 75% [3] and according to the availability of services in full-time organizations and firms in 24 hours (24/7) and having an disaster site in another location for accessing to services and their availability in difficult and natural disasters, which lead to failure and loss of datacenter in main location, is also an important matter. The rapid growth and the distributed sites of the datacenters increase the complexity of control and management processes. A new paradigm which is called Software Defined Systems (SDSys) comes as a solution to reduce the overhead of datacenters management by abstracting all the control functionalities from the hardware devices and setting it inside a software layer. These functionalities are responsible for organizing and controlling the main blocks of the datacenter; network, storage, compute and security. The Software Defined Datacenter (SDD) integrates the software defined concepts into all of these main blocks [4]. Data replication is one of the main mechanisms used in data grids whereby identical copies of data are generated and stored at various distributed sites to either improve data access performance or reliability or both [5], a dynamic data integrity checking mechanism in which the proof of correct data possession can be made from server on demand. RSA cryptosystem and homomorphic property, in combination, provide an effective method for generating the proof of correct data possession from the server, which its next aim would be to achieve data dynamics at a reduced cost in future [6].

The main idea in this paper is that, before placing the data on the server and checking its integrity, by using our proposed algorithm into which the simulated environment of SDDC's payload cluster is implemented; this can be shown that dynamic dividing of data in SDDC based on the type of data's importance (which are provided by service providers), and dynamic encryptions which are coincident with the proper policies and standards, lead to reduction in network traffic, improving in speed of data transmission (based on reduced volume of encrypted data), reduction in bandwidth cost, security increase for protecting data integrity, failover and fault tolerance in storing, efficient use of resources and storing with higher reliability.

2. RELATED WORK

Reducing network maintenance costs in firms and organizations which provide network services is an eternal desire and software-defined data centers are no exception. Wadha et al studied remote data integrity and the protection of integrity of data [6]. In [7] an algorithm is suggested for SDDC implementation which improves reliability in storing systems without adding new hardware [8], reducing latency to service the requests and throughput utilization which is studied and implemented in [9]. In the article [10], researchers proposed an algorithm to reduce energy saving for SDDC's network and increase the quality of services. However, in none of these articles the use of dynamic data clustering for reducing the bandwidth or volume of transferred data had been studied. In [6] the maintenance of cloud data integrity had been studied but the challenge of this article is that achieving data dynamics at a reduced cost. We proposed a new algorithm which divides and encrypts service data of service providers, dynamically (based on the type of data's importance), and sends them to separate storage parts. Also it relies on maintenance of data integrity, and in addition, it reduces encryption time, improves reliability, doesn't need new storing hardware and reduces traffic and bandwidth costs. In this article, many implementation methods had been compared and the relative fault tolerance approaches for different scenarios had been studied.

3. PROPOSED ALGORITHM

In order to apply this algorithm, we transferred available data in each of service providers to the web server based on their important type. After receiving the data which has been indexed with High, Normal and Low, are transferred to their separate storing location through RSA, DES and without encryption, respectively. Web server performs RSA algorithm after receiving more important data (which have High index) and connect to relative IP in its storing location, and send the encrypted data to that location. If received data has Normal index, DES algorithm is performed and connected to relative IP and send encrypted data to its storing location. And finally, if data has Low index, are sent to integrate (main) storage part, without encryption, after connecting to relative IP.

Here we create a data separating mechanism through dynamic filtering of transferred data from each service provider which divide them in contrast

with current methods for data transmission to main storage [6][7] which is shown in table 1. Figure 1 shows an overall view of proposed algorithm.

Table 1: priority, characteristics, and location of storing service provider data

No	Type of Service for Virtual Machine	Type of Priority in Simultaneous Transmission	Storage Machine
1	More Important services	High	More Important storage
		Low	Main (aggregation storage)
2	Less Important services	Low	
		Normal	Less Important storage

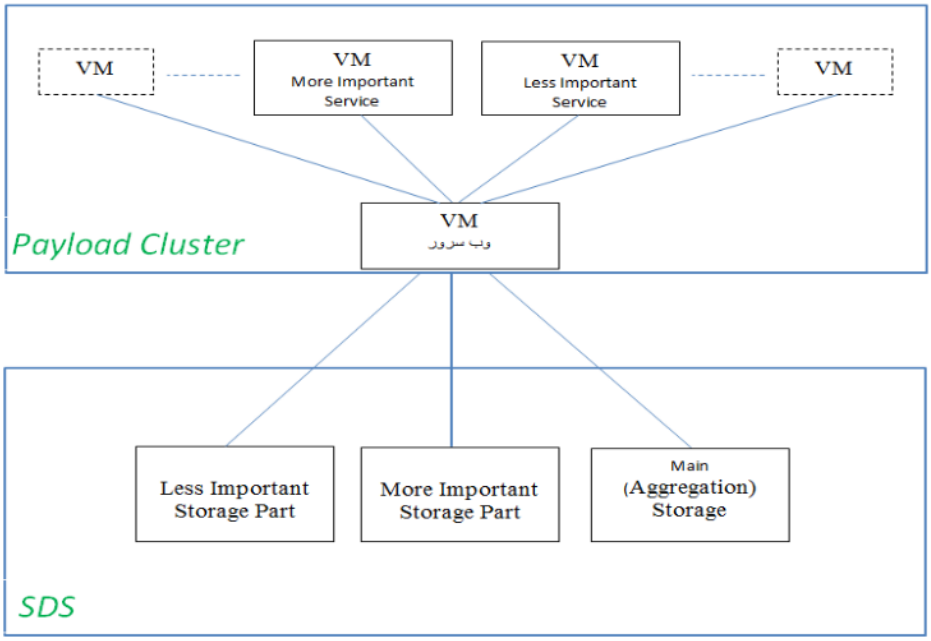

Figure 1. The conceptual view of proposed algorithm's implementation

Our proposed algorithm consists of the following:

Begin
Get data from VM(s)//VM: service content that provide service
Check type of data (High /Normal /Low)
If (is important (High))
Set encrypt RSA for data
Connect to network 1
Send to network 1
Else IF (is unimportant (Normal))
Set encrypt DES for data
Connect to network 2
Send to network 2
Else
Set data// that has a Low index
Connect to network 3
Send to network 3
END IF
END

At first, we drew network structure and requirements with full details, and in the next step the prepared structure was implemented in simulation software. We generated 6 virtual machines in VMware workstation 12 and the operating systems of each one, was installed with initial settings.

In the next step, we applied required configurations for RAM and CPU values in each of machines. In order to make required connections between these machines, IP settings in the similar range were generated in class C and the machines were installed in one network. The proposed algorithm is in one of the VMs with windows server 2008 which has an activated web server. According to the type of received data from more or less important service providers, if received data contain less important data which have Normal index, DES algorithm is executed on them and then transferred to less important storage part. If transferred data are more important which have High index, RSA algorithm is executed on them and the results are sent to more important storage part. And finally, the total of data which have Low index, without any encryption are sent to aggregated (main) storage part.

It should be mentioned that, two storages for storing the more/less important data, are used for data transmission toward disaster SDDC and aggregated

(main) storage part has no connection with outside of the organization. In tables 2&3 the characteristics of the system and their settings for simulation and implementation are shown.

Table 2. The characteristics and setting in implemented operating system

No	Winows	System Type	Memory	Processors
1	WIN 7 Ultimate Version 2009 Service Pack 1	64 bit	4 GB	Intel Core i5- CPU 1.80 GHz

Table 3. The systems characteristics and settings for proposed scenario

No	VMs	Winows	System Type	Memory	Processors
1	VM-More Important Service	WIN XP Professional version 2002 service pack 3	32 bit	512 MB	1
2	VM-Less Important Service	WIN XP Professional version 2002 service pack 3	32 bit	512 MB	1
3	Windows Server 2008	WIN Server 2008 R2 Enterprise	64 bit	800 MB	2
4	VM- More Important Storage	WIN XP Professional version 2002 service pack 3	32 bit	256 MB	2
5	VM- Less Important Storage	WIN XP Professional version 2002 service pack 3	32 bit	256 MB	2
6	VM-Main (Aggregation)Storage	WIN XP Professional version 2002 service pack 3	32 bit	128 MB	1

Sending data to storage parts of disaster SDDC is done through two less and more storage parts in the main site, and are transferred to two corresponding storage parts in disaster SDDC. Also, the main storage parts in these two centers have no connection with each other and are protected against a direct attack of attackers. Figure 2 shows a conceptual model of proposed algorithm in two software-defined datacenters.

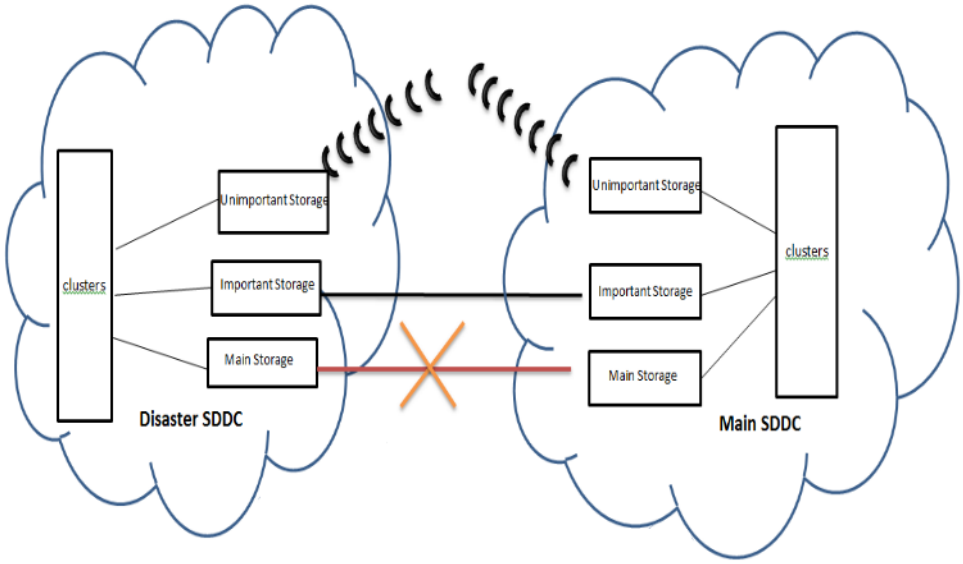

Figure 2. conceptual view of two software defined datacenters by applying the proposed algorithm

In this article, our existing data has been used from data set of Bank of England in 2011. These data are used for evaluating our methods. It should be mentioned that, these data have investigated in [11]. We selected data with different sizes to have a better-more accurate investigation. Then by using our proposed algorithm, register transferred volume, receiving data time from storage parts, are registered in comparative tables. And also, the diagrams and their network traffic are evaluated. We used the TXT files. Because, transmission and storing data as plain text makes them easier to view, maintain, increases their longevity, and/or makes it easier to share information with multiple languages and systems. In addition other benefits like: no special editor is required, accessible to your standard Power Tools,

relatively robust in the face of data corruption, there are standard libraries to parse XML, JavaScript, and so on, byte order issues are not a problem with ASCII Code or UTF Eight encoded Unicode [12].

We emulated the internal environment of payload cluster by using service provider machines in VMware workstation. And with implementation of proposed algorithm, we analyzed the receiving time of encrypted data in each of service provider machines. In addition, traffic control data are gathered and evaluated by Wireshark software, and related diagrams are extracted from the same software. Discussions about the reliability and usability of VMware workstation software from valid scientific circles proved its validity which has been used and referred to in [13-16]. In addition, the capabilities of Wireshark software for analysis of network traffic had been used in [17-20].

4. PROPOSED DESIGN

We considered 4 scenarios for implementation and test of proposed algorithm. In the first scenario, we executed RSA encryption on all of the data and transferred the data from one service provider to one storage part. In the second scenario, we executed RSA encryption on divided data in each service provider and analyzed their transmission to one storage part. In third scenario, we transferred the clustered and encrypted data to two separate storage parts, based on wrong encryption policy (symmetric instead of asymmetric and vice versa). In the fourth scenario (proposed algorithm), we transferred the clustered and encrypted data to two separate storage parts based on right encryption. It should be mentioned that, we transferred the data in each service providers without encryption to the main storage part In both scenarios 3 and 4. Finally we evaluated and compared these scenarios.

a. FIRST SCENARIO

In this scenario total volume is transferred together and from one service provider to main storage. Here there is no clustering based on important data type of service providers. Since more and less important data are mixed and applying security toward data integrity, service provider data with High index are sent to web server machine and after executing RSA, transferred to SDDC's main storage. In this scenario the only available storage is the main storage part in SDDC. Conceptual model of this scenario is shown in figure 3.

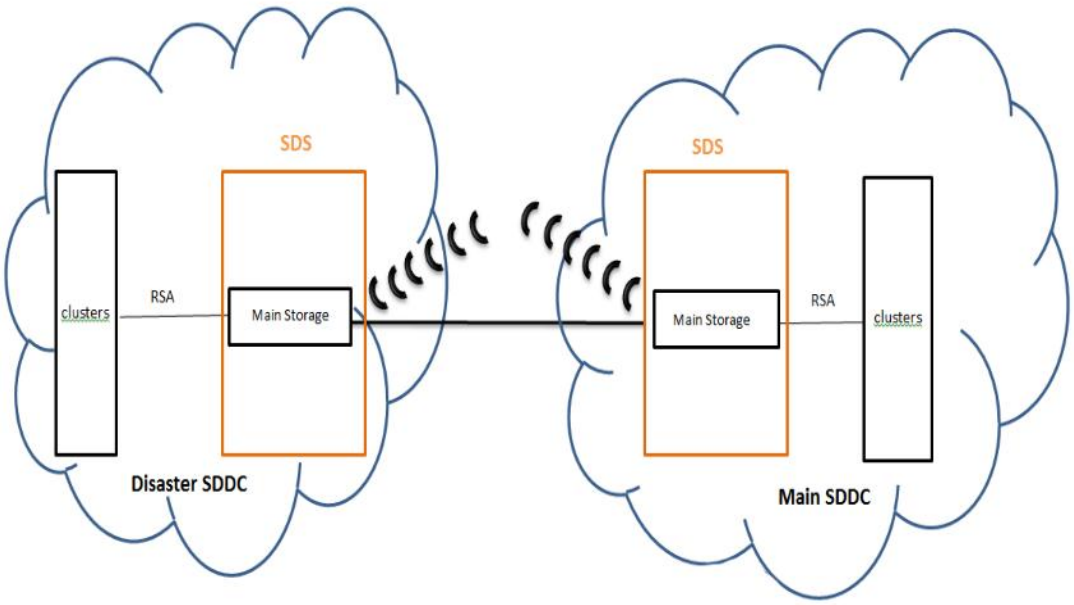

Figure 3. The conceptual view of the first scenario

After applying this process, the network traffic diagram for this scenario is shown in figure 4, receiving time for storing data, the volume of encrypted data and other characteristics are shown in table 4

.

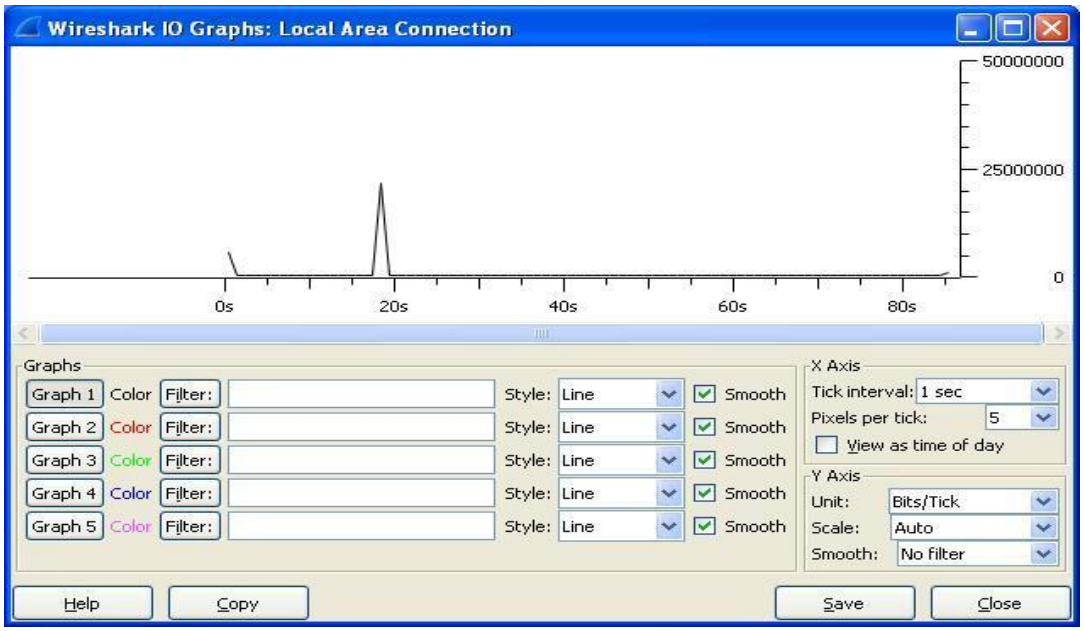

Figure 4. The network traffic diagram by sending data to main storage with RSA encryption

Table 4. The data and obtained results in first scenario

Type of Service	File Size (KB)	Total Transferred Volume from Service Provider (KB)	Type of Encryption	Maximum Network Traffic (bps)	Transmission Time (Sec)	Storage Part in Main SDDC	Volume in Storage Part in Main SDDC (KB)	Total Transferable Volume to Disaster Storage Part (KB)
Without priority	631	631	RSA	25,000,000	86	MAIN	2391	2391

b. SECOND SCENARIO

In this scenario, the data in each service provider which has High index, is sent to web server machine and after RSA encryption, is transferred to SDDC's main storage. In this scenario the only available storage is the main storage part in SDDC. And it should be mentioned that, the data transmission to this storage part is done, simultaneously. The conceptual model of this scenario is shown in figure 5. The network traffic diagram for this scenario is shown in figures 6&7. And also, receiving time in this storage part, the sizes of encrypted data and other characteristics are shown in table 5.

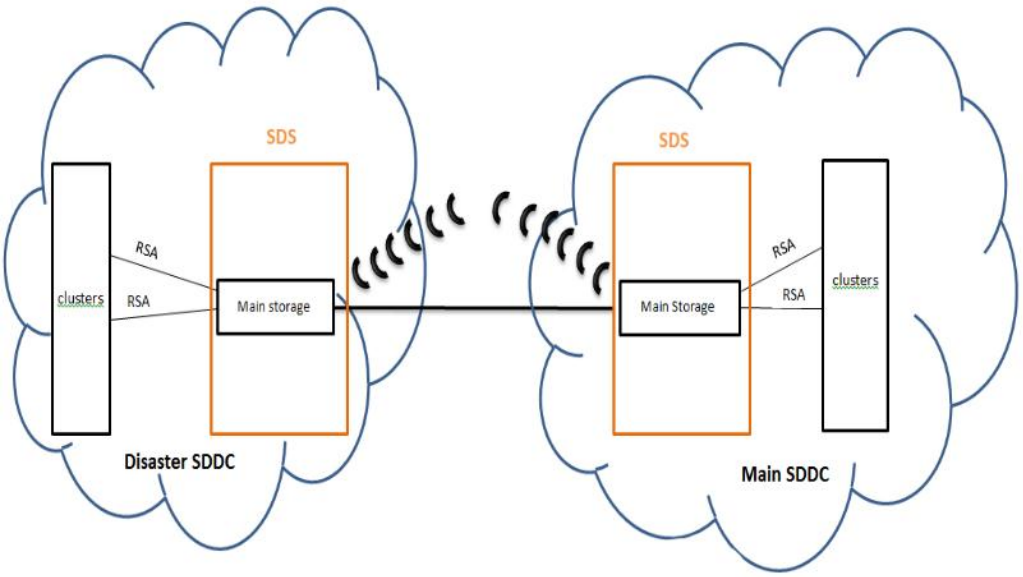

Figure 5. The conceptual view of second scenario

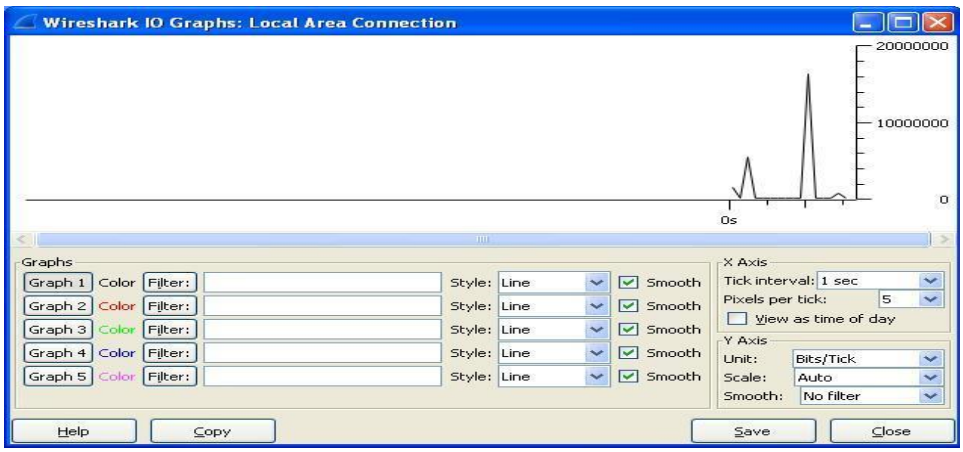

Figure 6. The network traffic diagram with simultaneous data transmission in more important service machine with RSA encryption

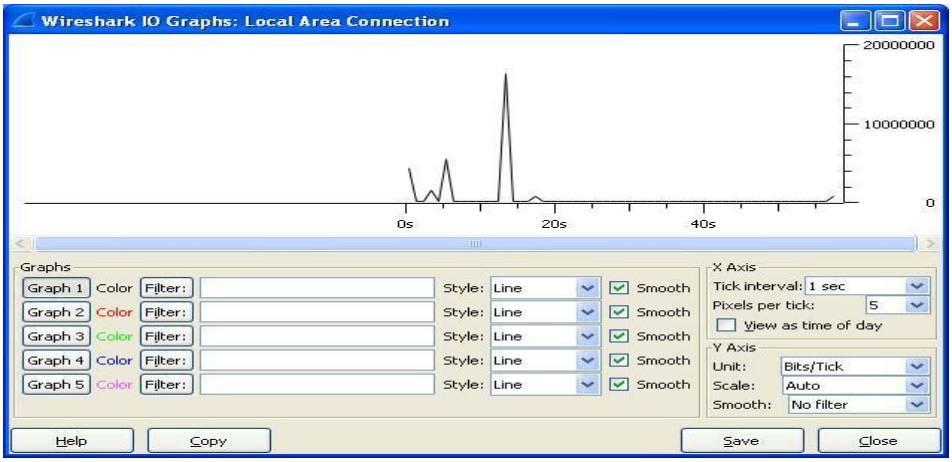

Figure 7. The network traffic diagram with simultaneous data transmission in less important service machine with RSA encryption

Table 5. The obtained data and results in second **scenario**

Type of Service	File Size (KB)	Total Transferred Volume from Service Provider (KB)	Type of Encryption	Maximum Network Traffic (bps)	Transmission Time (Sec)	Storage Part in Main SDDC	Volume in Storage Part in Main SDDC (KB)	Total Transferable Volume to Disaster Storage Part (KB)
More important	158	158	RSA	16,000,000	16	MAIN	598	2392
Less important	473	473	RSA	16,000,000	58		1794	

c. THIRD SCENARIO

In this scenario there are similarities with our proposed algorithm based on dynamic clustering in data storing but it has wrong policy in type of selected encryption (symmetric/asymmetric) technique. Here, the data of more important service providers which has Normal index, is transferred to web server machine and after DES encryption into the web server, is sent to more important storage part. The data of less important service provider which has High index, is transferred to web server machine and after RSA encryption, is sent to less important storage part. And the existing data in each of service providers with Low index are transferred to web server machine and without encryption is sent to main storage part. It should be mentioned that sending data to storage part, is done simultaneously. Conceptual model of this scenario is shown in figure 8. And also, network traffic diagram is shown in figures 9&10. The receiving time in these storage parts, the sizes of encrypted data and other its characteristics are shown in table 6.

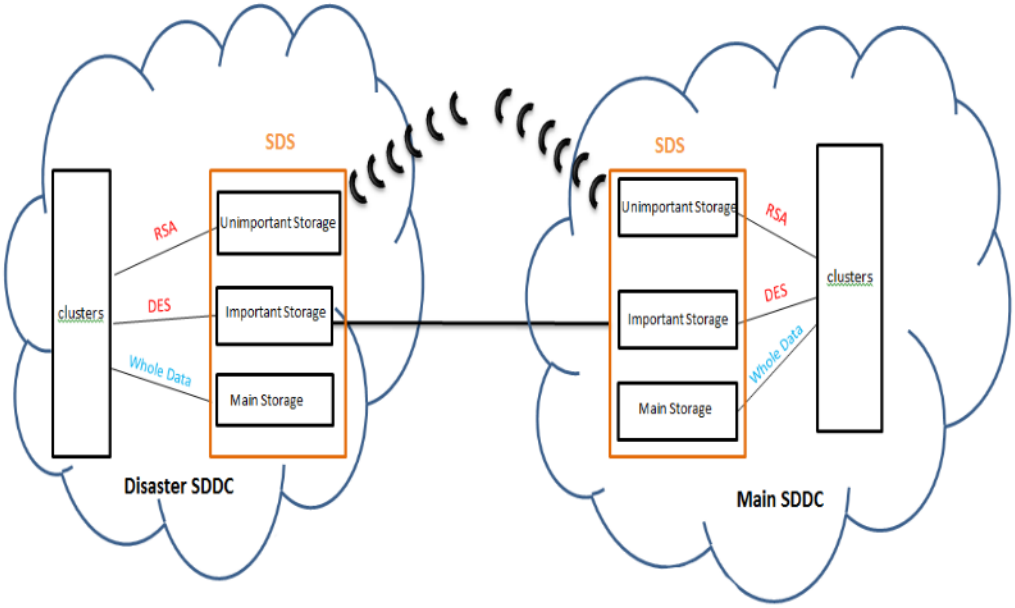

Figure 8. The conceptual view of third scenario

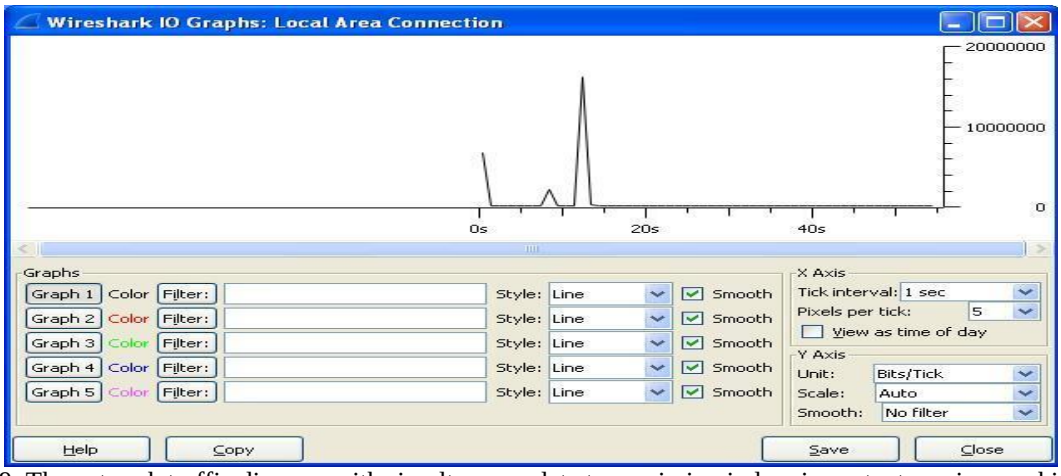

Figure 9. The network traffic diagram with simultaneous data transmission in less important service machine with RSA encryption

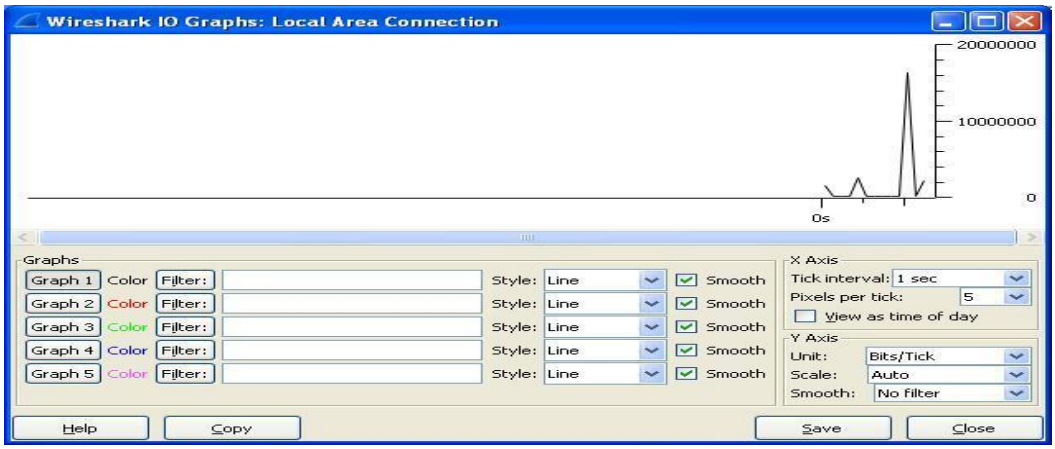

Figure 10. The network traffic diagram with simultaneous data transmission in more important service machine with DES encryption

Table 6. The obtained data and results in third scenario

Type of Service	File Size (KB)	Total Transferred Volume from Service Provider (KB)	Type of Encryption	Maximum Network Traffic (bit)	Storage Part in Main and Disaster SDDC	Transmission Time to Main SDDC's Storage Part (Sec)	Volume in Storage Part in Main SDDC (KB)	Total Transferable Volume to Disaster Storage Part (KB)
More important	158	316	DES	16,000,000	More important	12	1794	1794
Without priority	158		Without encryption		Main		158	
Less important	473	946	RSA	16,000,000	Less important	56	240	240
Without priority	473		Without encryption		main		473	

d. FOURTH SCENARIO (IMPLEMENTATION OF PROPOSED ALGORITHM)

This scenario is generated by our proposed algorithm. It has dynamic encryption which is implemented based on service type, appropriate and standard policy of encryption (symmetric/asymmetric) based on type of data (more/less important). In this scenario the data from more important service provider with High index, are sent to web server machine and after RSA encryption, transferred to more important storage part. Also the data from less important service provider are sent to web server machine (with Normal index) and after encryption with DES, transferred to less important storage part. And the existing data in each of service providers with Low index are transferred to web server machine and without encryption, is sent to main storage part. It should be mentioned that sending data to storage part, is done simultaneously. The conceptual model for this scenario is shown in figure 11. And also, network traffic diagram is shown in figures 12&13. The receiving time in these storage parts, the sizes of encrypted data and other its characteristics are shown in table 7.

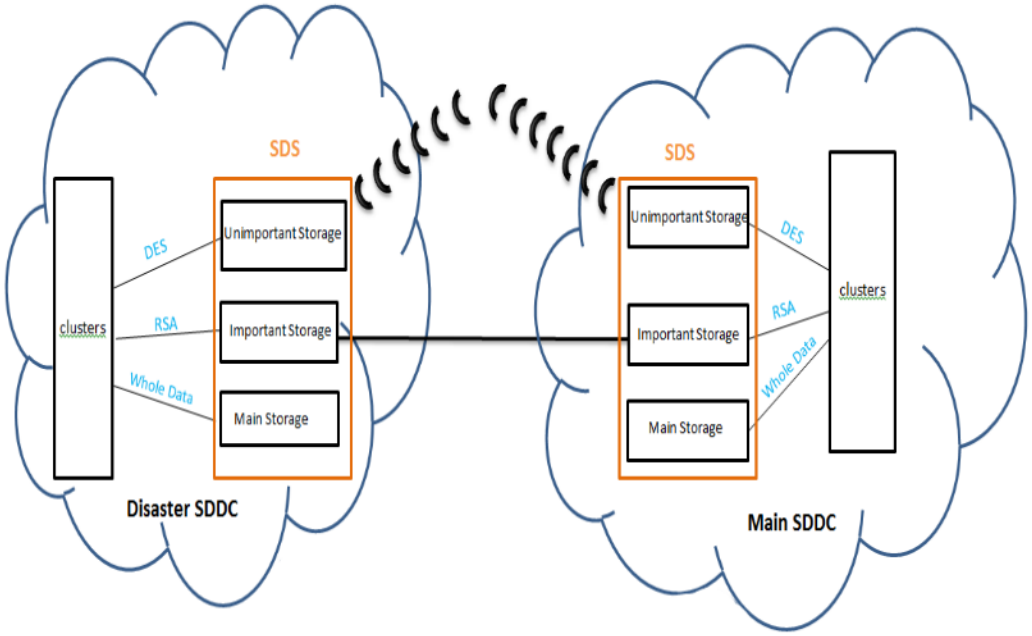

Figure 11. The conceptual view of fourth scenario

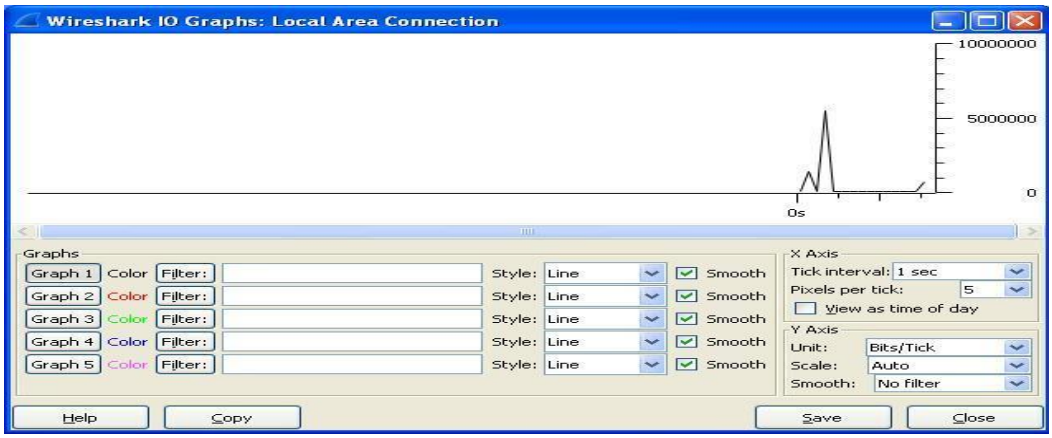

Figure 12. The network traffic diagram with simultaneous data transmission in more important service machine with RSA encryption

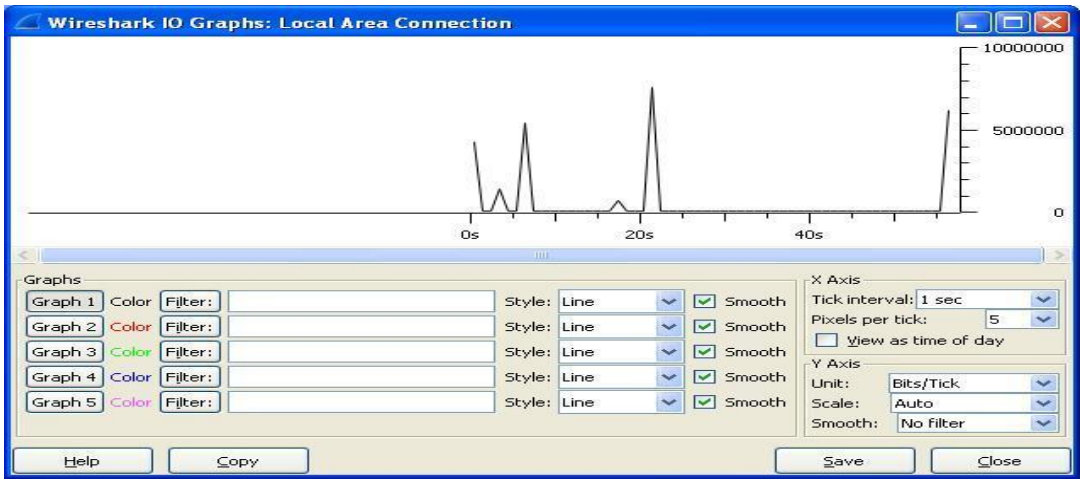

Figure 13. The network traffic diagram with simultaneous data transmission in less important service machine with DES encryption

Table 7. The obtained data and results in fourth scenario

Type of Service	File Size (KB)	Total Transferred Volume from Service Provider (KB)	Type of Encryption	Maximum Network Traffic (bit)	Storage Part in Main and Disaster SDDC	Transmission Time to Main SDDC's Storage Part (Sec)	Volume in Storage Part in Main SDDC (KB)	Total Transferable Volume to Disaster Storage Part (KB)
More important	158	316	RSA	5,500,000	More important	16	598	598
Without priority	158		Without encryption		main		158	
Less important	473	946	DES	7,500,000	Less important	56	719	719
Without priority	473		Without encryption		main		473	

e. EXPERIMENTAL OUTCOMES

In this article, Microsoft excel 2010 was used for statistical analysis. After inserting data from these tables in excel, we extracted required diagrams which show results. Figure 14 shows network traffic in comparison to fourth scenario and its differences with this scenario.

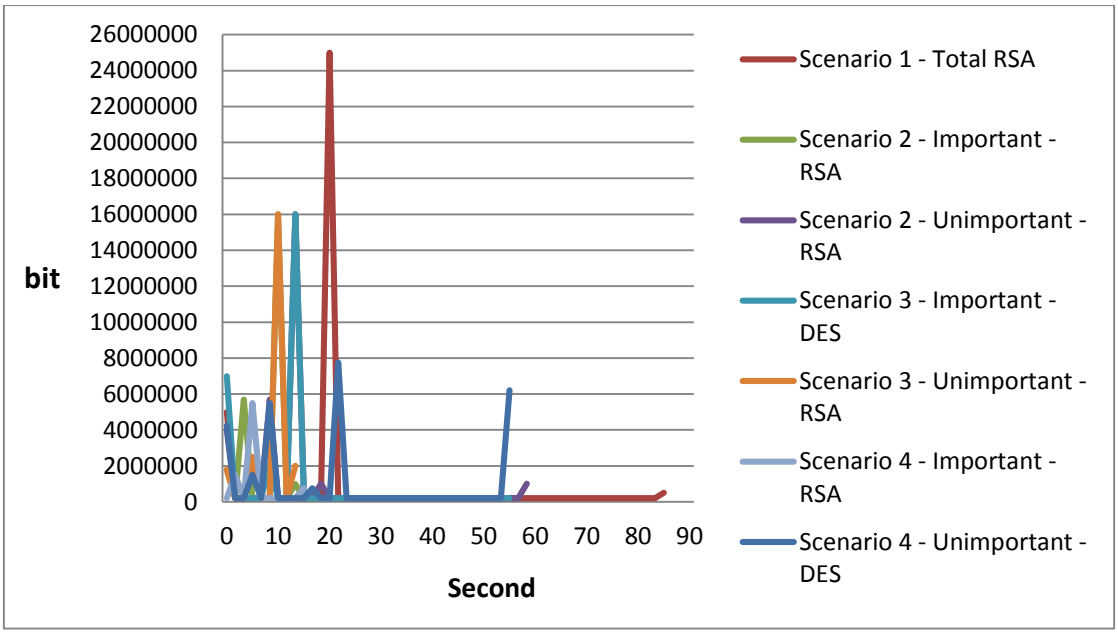

Figure 14. The network traffic in 4 scenarios

In addition, comparative diagram of encrypted data for transmitting to disaster SDDC's storage parts is shown in figure 15 which shows size reduction in fourth scenario.

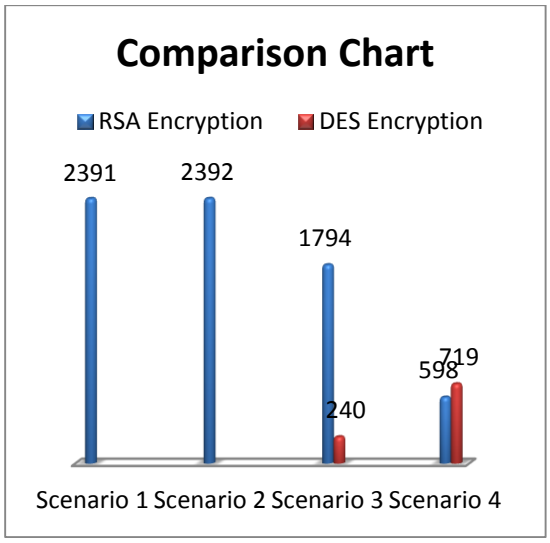

Fig 15. The comparison of sizes of encrypted data volumes in 4 scenarios

According to transferred data reduction to 25% from total volume in more important storage part and 30% of total volume in less important storage part, total amount of encrypted data which are transferred to disaster SDDC's storage parts reduced by 55% which improve the speed of data transmission and elimination of delays in data transmission and receiving it.

Since each organization has at least 2 transmission media to meet necessary redundancy (based on tier 3) and requirements in its data center. This, ensures service availability and reliability, according to 55% reduction in size of transferred data volume to disaster SDDC and applied clustering method by proposed algorithm (which can be seen in figure 15), there will be no need to prepare two similar high bandwidth in both of transmission media; and bandwidth purchasing is done, according to transferred and clustered data volume. In this process a transmission medium with higher bandwidth and another with lower bandwidth are purchased. This reduces the cost purchase of bandwidth.

With assumed values in tables 8&9, we evaluated and compared the required bandwidth in common method with proposed method. With this, we proved a 50% reduction in proposed algorithm mathematically.

Table 8. The required bandwidth for data transmission in common method

Total Data (without clustering) (MB)	Bandwidth in Two Transmission Media (mbps)		Total Bandwidth in Two Transmission Media (mbps)
1	10	10	20

Here, let d be total data, w1& w2 be bandwidth for each transmission media. And let S be total bandwidth for transmission media, we calculate total bandwidth in table 7 by equation 1.

We would always have: d <= S

S=W1+W2→W1=W2→ **S=2W** (1)

S=2*10→S=20 mbps

Table 9. The Required Bandwidth for Data Transmission in Our Proposed method

No	More/Less Important Data (kb)	Less/ More Important Data (kb)	Bandwidth in two transmission media (mbps)		Total Bandwidth in Two Transmission Media (mbps)
1	500	500	5	5	10
2	600	400	6	4	10
3	700	300	7	3	10
4	800	200	8	2	10
5	900	100	9	1	10

Here, we let d1 be more importat data, d2 be less important data. And let W1&W2 be bandwidths of transmission media, let S be total bandwidth of two transmission media which is shown by using equation 2 and we calculated total bandwidth in row three of table 8 (as an example)

We would always have: d1 <= W1, d2 <= W2

W1 ≠ W2 → **S=W1+W2** (2)

S= 7 + 3 →S= 10 mbps

For calculating S in each row, the answer is 10 megabit per second. Thus, total bandwidth is reduced by 50% in comparison to aggregate data transmission. So the cost of bandwidth purchase is reduced by 50%.

In first and second scenarios, certain security policies which are selected by organizations, lead to lack of dynamism in data encryption based on their importance (more/less important). And have a direct impact on type of selected encryption algorithm, the large volume of transmitted data, the type of transmission media, the purchased space bandwidth, and the consumed resources like CPU,RAM (according to generated traffic in network). As it was shown in third scenario, the wrong selection for type of data encryption in addition to possible dangers for organization's data, which increases of the use RAM, CPU, cost of bandwidth purchasing and encryption of unnecessary data.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new method for storing in software defined data-centers, in order to show the efficiency of this algorithm we proposed 4 scenarios. And by comparing and investigating obtained results in these scenarios, it was shown that executing the proposed algorithm led to network traffic reduced by 74.35% in comparison to first scenario , 60% reduction in comparison to second and third scenarios. The amount of volume reduction of transmitted data to disaster center which are stored there based on type of importance, it was shown that volume reduction was 55% in comparison to first and second scenario and 64.9% reduction in comparison to third scenario. Also, with respect to total data encryption time, the proposed algorithm has 14 seconds reduction in comparison to first scenario, 2 seconds reduction in comparison to 2nd scenario and 4 seconds increased in comparison to third scenario which this can be neglected because third scenario has invalid policy in contrast with right/recommended policies for data encryption and doubled network traffic time. By executing the propose algorithm in fourth scenario, reduced the volume of encrypted data. This increases the speed of data transmission. In addition, with the created reduction in network traffic, is helped to the load of the local network traffic. Also, we proved mathematically that reduction in size of prepared data for transferring in storage parts, reduced purchased bandwidth by half.

By executing our proposed algorithm and cutting the connection of WAN between main storage part in SDDC with corresponding storage part in disaster center in network (in contrast with the common method). This has prevented from direct attack of attackers to these two storage parts with the aim of elimination or distortion in main and precious data. In addition the provided security for maintaining the integrity would be improved. Implementation of this algorithm gave SDDC right redundancy to save data Because of storage parts that tier 3 technique is applied in a dynamic manner which improve availability and reliability for non-stop services. This improved fault tolerance for overcoming the failure with a safe and dynamic approach, and dynamic reliability in tier 3. Also we gave managers of organization the authority to select the right type of transmission media (cable/wireless) according to type of importance in transmitted data.

Our proposed algorithm executed based on data dynamism in storing process. We reduced the volume of prepared data for transmission and also increased the speed of transmission in order to the availability, maintenance of data integrity and reduction of the cost of bandwidth purchasing for data

transmission in this center, which are small but effective steps in this areas. In addition to many advantages which are available for this type of centers we improved dynamic data storing (based on fault tolerance), increased reliability and higher availability of services. Our study naturally suggests a course of exciting future work. We will investigate in clustering and storing the data, based on data contents of service providers which have combined with more and less important data, by using artificial intelligence algorithms.

6. REFERENCES

1. M. F. Habib, M. Tornatore, M. De Leenheer, F. Dikbiyik, and B.Mukherjee. (2012), "Design of disaster-resilient optical datacenter networks," IEEE/OSA Journal of Lightwave Technology, vol. 30, no. 16, Aug.
2. F. Dikbiyik, A. Reaz, M. De Leenheer, and B. Mukherjee. (2012), "Minimizing the disaster risk in optical telecom networks," in Proc. OFC/NFOEC, Mar.
3. Eric Ledyard. (2013), "SDDC is Here and Now: A Success Story" Vmware Inc.
4. Ala' Darabseh, Mahmoud Al-Ayyoub, Yaser Jararweh, Elhadj Benkhelifa, Mladen Vouk, and Andy Rindos. (2015), "SDDC: A Software Defined Datacenter Experimental Framework", 2015 3rd International Conference on Future Internet of Things and Cloud.
5. Jemal H. Abawajy, Senior Member, IEEE, and Mustafa Mat Deris, Member, IEEE. (2014),"Data Replication Approach with Consistency Guarantee for Data Grid ", IEEE TRANSACTIONS ON COMPUTERS, VOL. 63, NO. 12, DECEMBER.
6. Divya Wadhwa and Poonam Dabas. (2014), "A Coherent Dynamic Remote Integrity Check on Cloud Data Utilizing Homomorphic Cryptosystem", 5th International Conference-Confluence The Next Generation Information Technology Summit.
7. Ulya Bayram and Eric W.D. Rozier, Dwight Divine, Pin Zhou. (2015), "Improving Reliability with Dynamic Syndrome Allocation in Intelligent Software Defined Data Centers", 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks.
8. Habib Allah Khosravi, Mohammad Reza Khayyambashi. (2014), "Load-Aware Virtual Network Service over a Software Defined Data Center Network", 7th International Symposium on Telecommunications.
9. Akinniyi Ojo, Ngok-Wa Ma, Isaac Woungang. (2015), "Modified Floyd-Warshall Algorithm for Equal Cost Multipath in Software-Defined Data Center", Workshop on Advances in Software Defined and Context Aware Cognitive Networks.
10. Bo-Yu Ke, Po-Lung Tien, and Yu-Lin Hsiao. (2013), "Parallel Prioritized Flow Scheduling for Software Defined Data Center Network", 14th International Conference on High Performance Switching and Routing.
11. Philip Bunn and Lizzie Drapper. (2015), "The potential impact of higher interest rates and further fiscal consolidation on households: evidence from the 2015 NMG Consulting Survey" published in the 2015 Q4 Bank of England Quarterly Bulletin
12. JamesH.Armistead.(2014),"Power Of PlainText" http://wiki.c2.com/?PowerOfPlainText
13. Meryeme Ayache, Mohammed Erradi and Bernd Freisleben. (2015), "Access Control Policies Enforcement in a Cloud Environment: Openstack", 11th International Conference on Information Assurance and Security.
14. Michael Fry, Christopher Druzgalski. (2015), "Virtual Machine Based PFT and New Approach for Serialized Collaboration in the Clinic/Specialized Test Facility", PAN AMERICAN HEALTH CARE EXCHANGES (PAHCE).CONFERENCE, WORKSHOPS, AND EXHIBITS. COOPERATION / LINKAGES.
15. R. Mohtasin1, P.W.C. Prasad1, Abeer Alsadoon1, G. Zajko1, A. Elchouemi2, Ashutosh Kumar Singh3. (2016), "Development of a Virtualized Networking Lab using GNS3 and VMware Workstation", IEEE WiSPNET 2016 conference.
16. Henry Novianus Palit. (2016), "Deploying an Ad-Hoc Computing Cluster Overlaid on Top of Public Desktops", 8th IEEE International Conference on Communication Software and Networks.
17. Ashaq Hussain Dar, Beenish Habib, Mrs. Farida Khurshid, M. Tariq Banday. (2016), "Experimental Analysis of DDoS Attack and it's Detection in Eucalyptus Private Cloud Platform", Conference on Advances in Computing, Communications and Informatics, Sept. 21-24, 2016, Jaipur, India.
18. Deepak Vohra, Arusha Dubey, Khyati Vachhhani. (2016), "Investigating GSM Control Channels with RTL-SDR and GNU Radio", IEEE WiSPNET 2016 conference.
19. Abhijeet Desai, Nagegowda K S, Ninikrishna T. (2016), "An Approach To Efficient Network Design And Characterization Using SDN and Hadoop", International Conference on Circuit, Power and Computing Technologies.

20. IThato Solomon, 2Adamu Murtala Zungeru, 3Rajalakshmi Selvaraj. (2016), “Network Traffic Monitoring in an Industrial Environment”, IEEE.