Characterizing Storage Traffic of Virtual Desktop Infrastructure on Enterprise Network

Chunghan Lee* Naoto Fukumoto* Hiroshi Endo* Mariko Sugawara* Tatsuma Matsuki* Tatsuo Kumano*

* Fujitsu Laboratories Ltd., Kawasaki, Kanagawa 211-8588, Japan
{lee.chunghan, fukumoto.naoto, endo–hiroshi, sugawara.mariko, matsuki.tatsuma, kumano_tatsuo}@jp.fujitsu.com

Abstract—**Virtual desktop infrastructure (VDI) has been emerged in cloud computing, and commercial VDI systems have been widely adopted and deployed in the industry. In VDI, remote storages are used for user's virtual desktop, and it can be one of performance bottleneck points. Thus, it is important to understand traffic characteristics on storage area network (SAN) for improving VDI performance. However, a few studies have investigated their traffic characteristics. We explore the traffic characteristics of VDI on enterprise network. On the VDI, most of applications are MS office, anti-virus, web browsers, PDF readers, and so on. We thus amass actual FC traffic at a part of VDI systems in Japan for office work during a business day, and the dataset consists of roughly 1.7 billion packets. We report detailed views of FC traffic and patterns based on read/write transactions. We find that the read and write are approximately 76.3% and 23.7%, respectively. The similar proportion is found from logs on storages. Moreover, approximately 50% are 4KB, 80% are smaller than 20KB, and 99% are smaller than 128KB. Although only 20% are larger than 20KB, the ratio in total bytes is close to 80% in both the read and write. The characteristic is similar to that of mice and elephant flows on datacenter networks even if the types of applications and communication patterns are different from datacenter.**

Index Terms—**Virtual desktop infrastructure (VDI), Storage area network (SAN), Enterprise Network, Storage, FC, FCoE**

I. Introduction

The cloud computing has been widely used in the world. Virtual Desktop Infrastructure (VDI) [1][2] is a type of cloud computing. It can reduce hardware replacement costs, save money in power cost, manage many virtual machines (VMs) on centralized system easily. With these merits, VDIs are broadly deployed at universities, companies, and government offices.

In VDI, the modern network, storage, and virtualization technologies are adopted to accomplish the merits. Normally, virtual desktops are provided to users as VMs on remote servers. The users remotely connect to the VMs using their client PCs. Particularly, remote storages on storage area network (SAN), such as iSCSI or Fibre Channel (FC) [3] are also used to access data on VMs. If a user reads e-mail or opens a file, their data is transferred from the storage to VM on server. Thus, SAN is one of important factors related to the entire system performance of VDI. While there are many traffic analysis results on the Internet [4], enterprise networks [5][6], and datacenter network (DCN)s [7][8][9], a few studies have investigated storage traffic characteristics on VDI. The previous works [10][11] only focused on the evaluation of their proposed methods and used artificial data using benchmark programs. If we clarify the storage traffic characteristics on VDI, we could use the results for capacity planning. Moreover, it would be possible to provide different network services, such as application-specific QoS and network path, based on the storage traffic characteristics.

The aim of our study is to characterize the storage traffic of VDI on enterprise network. On the VDI, most of office business applications are MS office, anti-virus software, web browsers, PDF readers, and so on. Thus, such application is neither High-performance computing (HPC) nor MapReduce. We firstly used actual FC traffic data set captured at a part of VDI systems[1] in Japan. It consists of 1.7 billion packets during one business day. Second, we carefully extracted exchanges (R/W transactions) based on FCP headers from the data set. Third, we analyzed them using well-known metrics, such as exchange size and exchange transfer time. Finally, we also gathered logs recorded at servers and storages, and verified the analysis results in comparison with their logs. The main findings of our analysis are as follows.

- In 84 million exchanges, the write exchanges are approximately 23.7% while the read exchanges are approximately 76.3%. We also find the similar results from the logs on the storages.
- In both read/write exchanges, approximately 50% exchanges are 4KB and 80% exchanges are smaller than 20KB. However, the maximum size of read and write is increased to 531,456 bytes and 656,896 bytes, respectively.
- Although only 20% exchanges are larger than 20KB, the ratio in total bytes is close to 80% in both the read and write. This characteristic is similar to that of mice and elephant flows on DCNs even if the types of applications and the communication patterns are different from datacenter.

The rest of this paper is organized as follows. We firstly introduce VDI and target VDI system in Section II. Next, we present FC capture system at the target VDI and describe VDI monitoring system for logs at servers and storages in Section III. Then, we show analysis results of FC traffic on the VDI and they are compared to the logs for the verification of

[1]We cannot disclose the company name due to the contract with the company operating the VDI system.

our results in Sections IV and V. Finally, we describe related work and conclude with a summary of the main points in Sections VI and VII.

II. Overview of VDI

A. Virtual Desktop Infrastructure (VDI)

In offices, costs for security, management, and PC replacement are very high. In order to reduce them, VDI has been used at offices, universities, and business companies. It is one of desktop virtualization technologies, and a server-based computing that provides remote desktop environments. However, it is different from the ordinary remote desktop. In VDI, virtualization techniques are utilized to consolidate multiple desktop environments onto a single physical server and a remote disk at storage side is mounted to a VM. Thus, the desktop is provided to a user as a VM.

From hardware view, it consists of servers, storages, client PCs, storage area network (SAN), and high-speed network for LAN. Storages are used to store VDI data, and SAN connects VDI servers to the storages. Then, the network of Gb Ethernet or faster is required to access the remote VM from the client PCs. The overview of traditional VDI can be illustrated in Fig. 1. FCoE (Fibre Channel over Ethernet) is a reasonable way because it can provide an integrated network of LAN and SAN. Actually, the FCoE is adopted for target VDI.

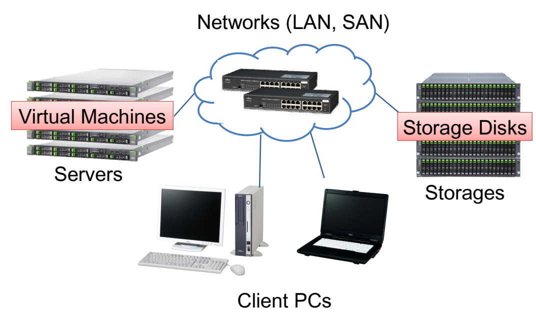


Fig. 1. Overview of VDI.

In VDI, various types of OS, such as Microsoft Windows XP, Microsoft Windows Server family, Windows 7/8, Linux, and so on, are provided for VMs. In order to provide such OS, a golden (master) image is used to deploy the VM. By using the golden image, the usage of storage can be reduced. The common data is stored in the golden image, and individual data per VM is stored in other blocks. For example, when a large number of VMs are deployed without the golden image, the required volume is proportion to the number of VMs. When the golden image is used, the required volume consists of the sum of the golden image and a small size of volume for the individual data.

There is a connection broker [2][12] to monitor resource and to connect/reconnect users to available VMs in VM pools or designated user VMs. It is also in charge of the automatic updates for virus and common software, such as windows updates. There are two types of management of the broker. The former is to use common VM pools. In this type, all of VMs are existed in the pools, only allowed applications can be installed, and the broker updates VMs commonly and simultaneously. Thus, the broker firstly downloads the related patches, and applies them to the VMs without the Internet. After the updates are completed, the VMs are provided to users in sequence. To summarize, the broker controls the all of VMs forcedly. The latter is to provide designated VMs for users without the pool. There are two major different points. The first one is that users can install their software freely. The second one is that the patches can be downloaded by individual VMs from the Internet. The different style of updates may lead to large overhead in the entire VDI system. Our target VDI for analysis is the latter case. We will show the overhead caused by the common operations, such as the update and virus scan in Section V.

B. Target VDI system

Fig. 2 shows the overview of target VDI with measurement system. The target VDI is configured with 6 VDI servers, 6 storages, and a FCoE fabric. Table I shows the specifications of server and storage. The VDI servers have the same configuration, i.e., Intel CPU E5-2695v2 (2.4GHz, 12core, 30MB) and 256GB of memory. ETERNUS DX90 S2 is used for the six storages whose HDD is 8TB (900GB SAS, 10krpm). Servers and LUN are statically mapped each other.

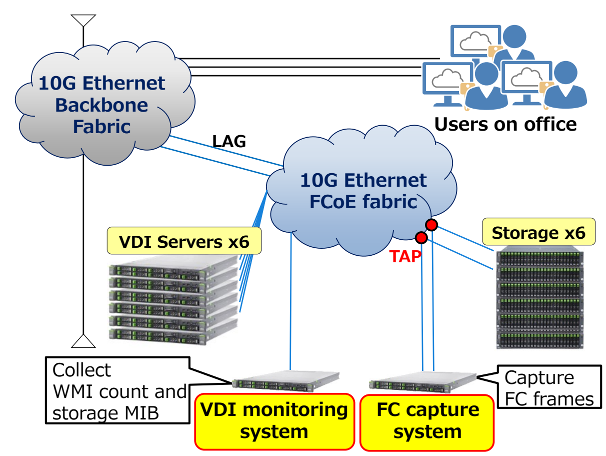


Fig. 2. VDI with measurement system.

In order to understand the application workloads, we investigate the types of applications. On the VDI, MS office (Word, PowerPoint, and Excel), mailer, PDF reader, anti-virus software, web browsers, text editors, and instance messenger are mainly used. Moreover, some users use an integrated development environment (IDE), such as visual studio, eclipse, and NetBeans, for their developments. To summarize, such application is neither HPC nor MapReduce, and there is no partition/aggregate communication pattern.

The users on the office connect to the VMs on the VDI servers through the network from the client PCs. Each VDI server hosts about 50 VMs. The types of OS on VM are consist of Microsoft Windows 7 32bit or 64bit. Applications, i.e.,

TABLE I
SPECIFICATIONS OF TARGET VDI SYSTEM.

Server	Server Fujitsu PRIMERGY RX200 S8	
	CPU	CPU Intel E5-2695v2 (2.4GHz, 12cores, 30MB)
	Memory	256GB
Storage	Fujitsu ETERNUS DX90 S2	
	HDD	HDD 8TB (900GB SAS x9, 10krpm)
	RAID	6
Virtual Machine	OS Win7 Enterprise 32bit or 64bit	
	Logical CPU	2 cores
	Disk	100GB
Network	Commodity FCoE 10G switch fabric	
VDI version	WS2008R2(Hyper-V) + XenDesktop 5.6	

IE, Firefox, Google Chrome, Microsoft Office, and Acrobat Reader, are mainly executed on the target VDI during the working hours. These applications are normally read-heavy in IOPS. In order to provide the OS, the golden image is provided to deploy the VM. In this VDI, the common data is stored in the golden image, and individual data per VM is stored in other blocks. The original size of the golden image is 50GB. The individual data is not preserved on servers, but is transferred and saved on the storages through the FCoE fabric that connects between VDI servers and storages. The FC capture system is connected to capture the FC packets between storages and FCoE fabric. The VDI monitoring system is also connected to the FCoE fabric to collect the information about the VDI servers and storages.

III. VDI MEASUREMENT SYSTEM

A. FC capture system

We captured packets between servers and storages. The packets are copied to a capture server using port-mirroring of network switch. The capture server stores the full payload as binary data and extracts part of headers in binary data using Tshark [13]. We carefully selected FC [14] and FCP headers [15] to analyze FC traffic. *Fc.ox_id*, *fc.rx_id*, *fc.fctl.exchange_first*, and *fc.fctl.exchange_last* are used for separating the packets into exchanges. We considered *fc.relative_offset* of last data packet in an exchange as exchange size and calculated transfer time as the difference of time stamps between first and last packets in an exchange.

Table II shows hardware specifications of the capture server. We used the packet capture system [16] to monitor the 10Gb Ethernet network. For analysis, we captured the packets from 8:00 to 20:00. Approximately 0.1% of capture data were lost in the mirror port during the packet capture.

TABLE II
HARDWARD SPECIFICATIONS OF TRAFFIC CAPTURE SERVER.

CPU	Intel Xeon E5-2670 (2.6Ghz, 8cores)
Memory	256GB
HDD	15TB
Network card	Intel 82599 10GbE

B. VDI monitoring system

We built a monitoring system to monitor performance of VDI, which was configured with the Metrics collector and Database as illustrated in Fig. 3. These were implemented with *Fluentd* [17] and *MongoDB* [18]. The metrics collector fetches performance metrics from the VDI servers and storages in Fig. 2 and timestamps each metric. It also attaches labels to the collected data. The attached labels include information from which the metrics have been collected. Finally, it forwards the data to the database and all metrics are stored in the database.

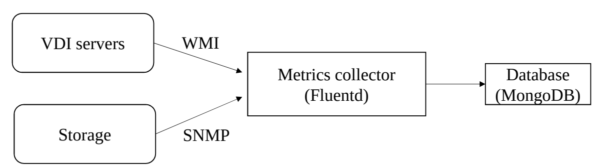


Fig. 3. VDI monitoring system.

We used simple network management protocol (SNMP) [19] to obtain the management information base (MIB) counters of the physical storage. The storage supports some private MIBs, which include metrics related to storage such as read/write response times, I/O per second, read/write throughput, and cache hits for each LUN. These metrics were mainly used to verify our analysis of FC packets as will be explained in Section V. We ran *snmpwalk* command on the Metrics collector with a fixed interval (60 s) and obtained them. We also used Windows management instrumentation (WMI) [20] for collecting metrics that are associated with the virtual disks of individual VMs on the target VDI such as *read/write bytes per second* and *read/write operations per second* [21]. Analyzing these metrics enables the disk utilization of individual VMs to be observed. We ran *wmic* command [22] on the metrics collector with a fixed sampling interval (60 s) and obtained these metrics.

Some obtained metrics have an incremental (cumulative) feature such as a read/write operation *counter* on a virtual disk. These metrics were calculated difference between successive data and the difference was divided by the sampling interval. These calculations were done on the metrics collector.

IV. ANALYSIS OF FC TRAFFIC ON VDI

A. Overview of FC/FCP protocol

FC is used for SAN and based on a structured architecture providing specifications from both physical layer and transport protocols. FCoE encapsulates the FC frame as the Ethernet frame and is able to integrate SAN and LAN on the same physical layer. A FC transport protocol provides facilities called exchange to identify and manage operation between an initiator and a target to send Information Units (IUs). Fig. 4 shows the structure of exchange on FC. In this figure, a *frame* (a packet in TCP/IP) is the briefest unit and has its own protocol specific information that must be delivered to another port to perform a transaction. FC protocol for SCSI (FCP) IU has four units as follows; Command (FCP_CMND) transfers the type of operation, i.e., read or write. At the beginning of a transaction, it is sent by an initiator. Data (FCP_DATA) transfers data between an initiator and a target. It is sent by both an initiator and a target. Transfer-ready (FCP_XREF_RDY) transfers an allocation of write data buffer size. It is sent by a target. Response (FCP_RSP) transfers the status of a transaction. If the status is normal, 'good' flag is marked. It is sent by a target. These IUs are transported by using a *sequence* that is composed of one or some frames. An *exchange* consists of one or more sequences that include related IU. We analyzed the dataset in units of exchange.

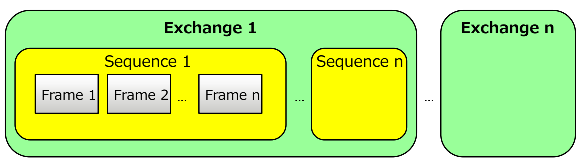


Fig. 4. Fibre channel exchanges.

SCSI read and write operation examples of FC are shown in Figs. 5 and 6. At first, an initiator sends FCP_CMND to a target. After that, in the case of read, one or more FCP_DATAs are transported. Finally, the initiator is received FCP_RSP with good status and the exchange is finished. As shown in Fig. 5, the exchange indicated in green is composed of three sequences indicated in yellow. In the case of write, after the FCP_CMND, the target sends a FCP_XREF_RDY. After that, as is the case in read, one or more FCP_DATAs are transported. If the size of data is large and the entire of data cannot be transported at once, a FCP_XREF_RDY is send again. After all data is transported, the initiator is received FCP_RSP with good status and the exchange is finished. As shown in Fig. 6, the exchange indicated in green is composed of six sequences indicated in yellow.

B. Dataset

We collected over a period of 12 hours from 08/24/2015 Mon 08:00 to 08/24/2015 Mon 20:00, and stored in approximately 5TB of disk. In order to capture packets with our FC capture system [16], a port-mirroring at the commodity FCoE 10G switch is used. Most of traffic at the ports that connects the storages is the FCoE traffic only. During the capture period, approximately 300 VMs are running on the target VDI. The total number of active VMs is calculated with WMI counters at the VDI servers.

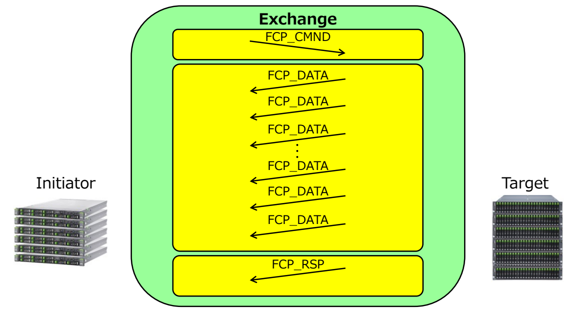


Fig. 5. SCSI read operation of FC.

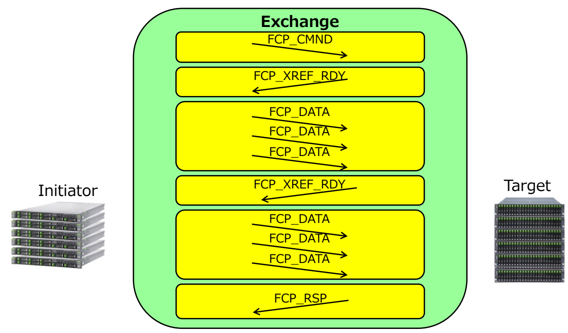


Fig. 6. SCSI write operation of FC.

In our dataset, the total number of FC packets is 1,768,175,589. From this dataset, we extracted exchanges 84,572,869. Moreover, the number of read exchanges is 64,547,015, and the number of write exchanges is 20,025,854. In the entire exchanges, the write exchanges are approximately 23.7% while the read exchanges are approximately 76.3%. We estimated two major causes of the above results. The first one is that there are no local disks on the VDI servers. The second one is the type of most frequently used applications on the target VDI. These business applications create a lot of read exchanges normally when the applications are launched and the related files are opened.

C. Read/write exchange size

The exchange size normally indicates the size of transaction. This size is similar with the flow size in TCP/IP. It relates to the buffer size on the initiator. In order to investigate changes in the size, we calculated the CDF values of exchange sizes (Fig. 7). Approximately 80% exchanges are smaller than 20KB. Particularly, we found that the 4KB has the largest ratio (approximately 50%) in both read and write exchanges. OS block size (4KB) would be a major cause of this result. However, we cannot clarify the correct reason why the 4KB has the largest ratio. In future, we intend to make clear the above reason to investigate types of applications on VMs. Furthermore, the ratio of 128KB is approximately 10%, and there are the large sizes, such as 256KB and 512KB. In the read size, the minimum, mean, maximum values are 2048 bytes, 199,967 bytes, 531,456 bytes, respectively. The write size is also similar to the read. However, the maximum size

(656,896 bytes) is larger than that of read.

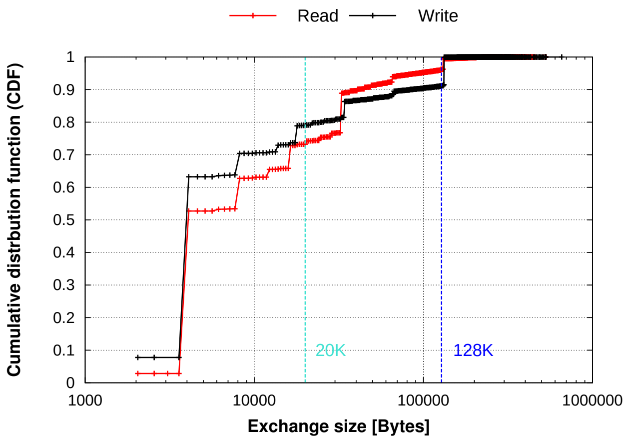


Fig. 7. CDF of read/write exchange size.

Our results are very different from those shown by the previous works [10][11]. Although VDI system for analysis is an office, our results are different from the results by Shamma et al. [10]. In their analysis results, the number of write exchanges is larger than the read. But, the write at ours is small in comparison with the read. Next, in [11], 4096 bytes is not the largest ratio at the read exchange. Moreover, there is no large exchange size above 128KB.

D. Read/write exchange transfer time

The exchange transfer time depends on many factors. The representative factors are allocated buffer size and the number of sequences in one exchange. The CDF values of read/write exchange transfer time are shown in Fig. 8. In our analysis, we limited the transfer time below 1 second. Due to the packet loss at the mirror port, 0.01% of the transfer time is a very large value, such as hundreds of seconds. A major cause of the large transfer time is the continuous packet loss at the mirror port. During the exchange extraction, there are two exchanges that have the same *fc.ox_id* and *fc.rx_id*, but the last packet (*fc.fctl.exchange_last*) at the first exchange is lost and the first packet (*fc.fctl.exchange_first*) at the second exchange is lost continuously. In this case, the first packet at the first exchange and the last packet at the second exchange are used to extract the exchange. As a result, the incorrect exchange is created. We eliminated all of the above cases from our analysis results.

Normally, a recommended threshold [23] of the transfer time is approximately 20 or 30ms. If the transfer time was larger than the threshold, users would have an experience with delayed response of FC. In our analysis, we found that some transfer times were larger than the threshold. While approximately 27.5% of the read are larger than 20ms, approximately 6.1% of the write are larger than the threshold. Moreover, the CDF values in the write are significantly different from those in the read. From this result, the CDF shape of the read is fluctuated in comparison with the write. A major cause of the fluctuation would be the basic characteristics of read/write operations. Thus, the read should wait a time for finding the correct data on remote disk while there is no the waiting time at the write. We will clarify the cause of fluctuated transfer time as a future work. Moreover, we should investigate what impact is a major cause of the large transfer time.

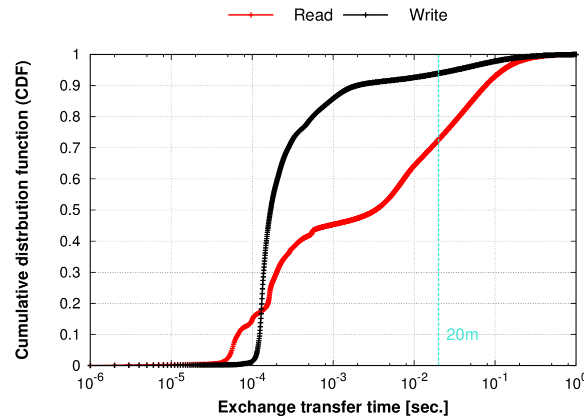


Fig. 8. CDF of read/write exchange transfer time.

E. Read/write exchange size on total bytes

Because the 4KB exchange had the largest ratio, we focus on how this exchange size affects the total bytes in the data set. The CDF values of read/write exchange sizes on total bytes are shown in Fig. 9. While the 4KB exchange is approximately 50% in the total number of exchanges, that is only approximately 10% in the total bytes. We can find this characteristic in both the read/write exchanges. On the other hand, the large exchange, such as 128KB, at the read/write has the large ratio. In particular, the 128KB exchange at the write is roughly 50.2%.

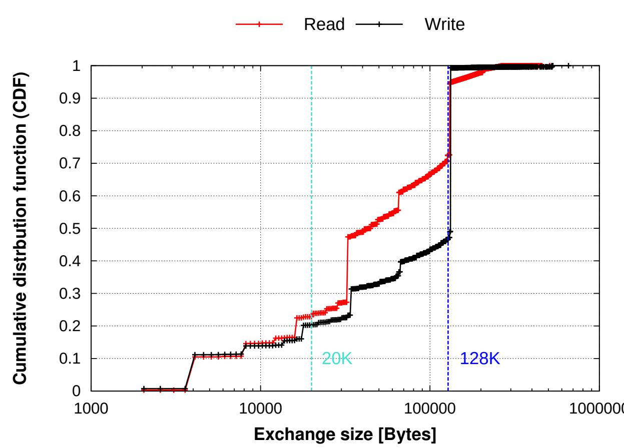
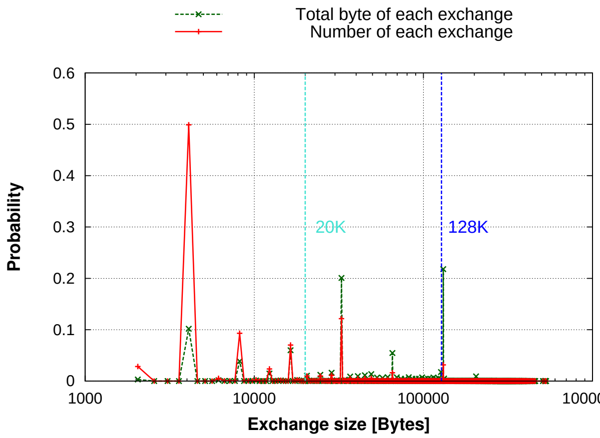
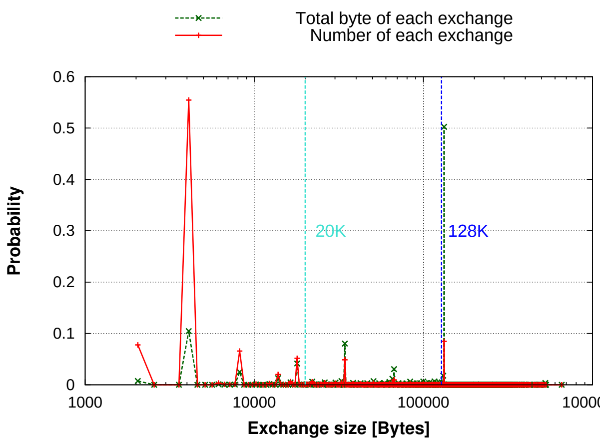


Fig. 9. CDF of read/write exchange size on total bytes.

The ratio of the total bytes is larger than that of the number of exchanges when the exchange size is greater than 20KB. The PDF of read/write exchange size on total bytes is shown in Fig. 10. Thus, it will be efficient to provide different network services, such as QoS, network path, and different pause mechanism for congestion control, based on the observed traffic characteristics. To summary, the characteristics are very similar to those [7][8][9] between mice and elephant flows on DCNs.



(a) Read



(b) Write

Fig. 10. PDF of read/write exchange size on total bytes.

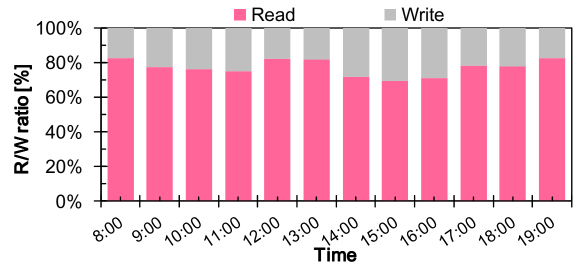
V. VDI Utilization on storages

A. Dataset

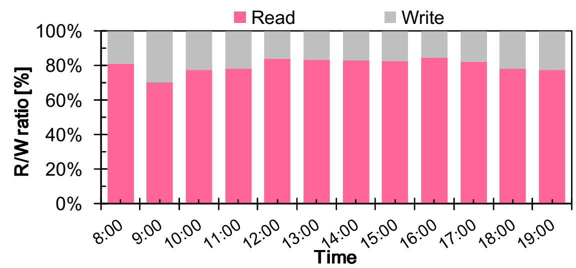
In order to verify our analysis using FC packets, we obtained the storage MIBs from VDI monitoring system over a period of 24 hours from 08/24/2015 Mon 00:00 to 08/29/2015 Sat 00:00. The period includes the day 08/24/2015 when the FC packets were analyzed as mentioned in Section IV. This dataset is obtained every minute, and we condense the data into the average values of each hour. We calculated the input/output per second (IOPS) that is the input/output counts per second into an hour. Using the IOPS, the exchange counts are verified on not only FC traffic as mentioned Section IV, but also storage side. We also calculated the average throughput per an hour.

B. Storage performance for 1 day

We verify the analysis results in section IV-B using storages MIBs. Fig. 11 shows the ratio of read and write IOPS counts obtained from the MIBs for the 12-hour period. We calculated IOPS and throughput per 1 hour. These graphs show that read IOPS and throughput account for 80%. These results correspond with roughly FC analysis results in section IV-B.

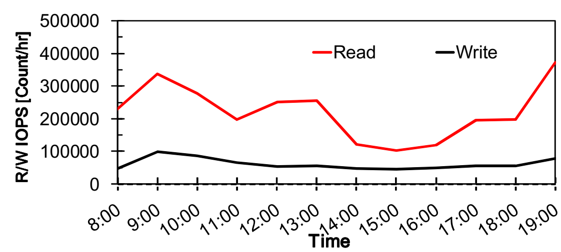


(a) IOPS

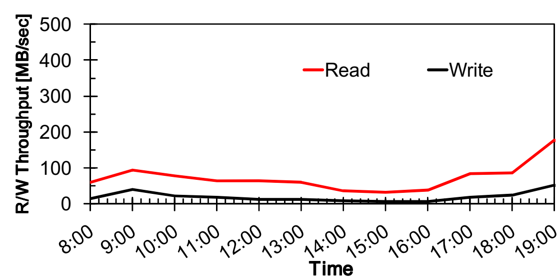


(b) Throughput

Fig. 11. Ratio of IOPS and throughput on 08/24/2015.



(a) IOPS



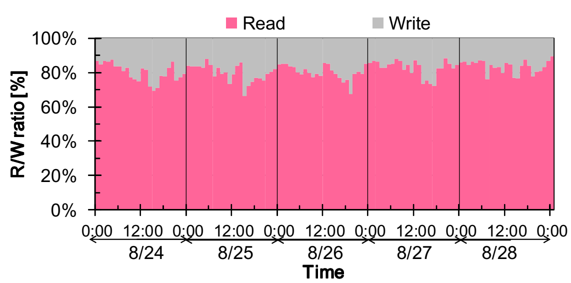
(b) Throughput

Fig. 12. IOPS and throughput on 08/24/2015.

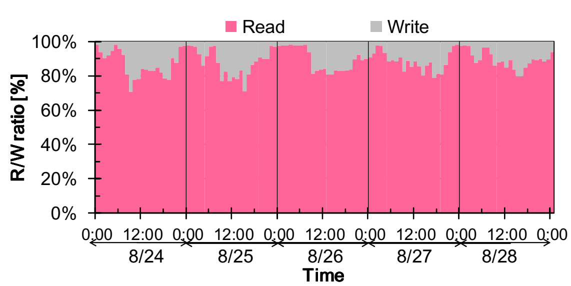
Fig. 12(a) plots the absolute value of IOPS for the 12-hour period. The IOPS increases between 8:00 and 9:00 when the working starts. Other peak is observed in 19:00, which is outside working hours. This is caused by updates of antivirus software and simple scans in all VMs at nearly the same time. We investigated the above cause using antivirus software logs. Fig. 12(b) shows throughput for the 12-hour period. Throughput value is low in 10Gb FCoE network even in the medium scale VDI system (300 VMs). We inferred that there are few read and write requests for large files in the target VDI. The target VDI used for office work (not used HPC or data intensive application), there are few read and write request to large file. Furthermore, large file requests are divided into the small size exchanges. The results demonstrated in Section IV-C also show that the small size exchanges account for a major portion of communication pattern. This communication pattern is very differs from TCP/IP. When a large file is transmitted in TCP/IP, a single flow with a large congestion window is normally used to achieve high throughput without network congestions. In future, we will precisely analyze how the large file requests are separated into exchanges.

C. Storage performance for 5 days

We also used statistical information of storages for 5 days to verify the results in Fig. 11 during long term. Fig. 13 shows the ratio of IOPS and throughput from 08/24/2015 to 08/28/2015. We also observed that the similar percentages of IOPS and throughput. On the other hand, we observed that there is a time when the ratio of read throughput is approximately 90% from the results in Fig. 13(b). This reason is that about half of VM data is transmitted from VDI storages to backup storages among 8:00 PM to 5:00 AM.



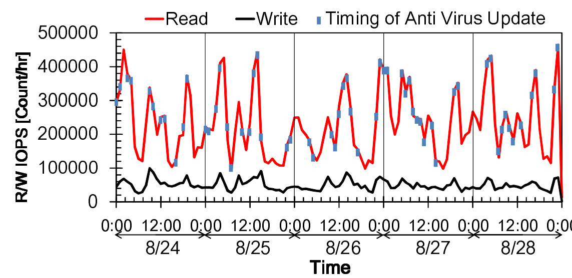
(a) IOPS



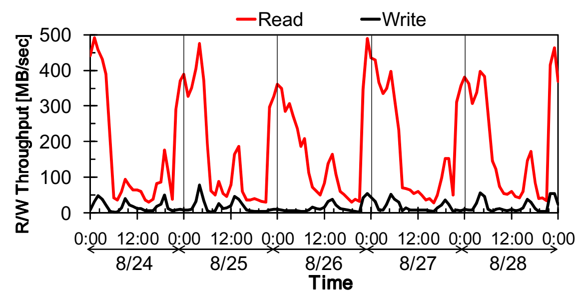
(b) Throughput

Fig. 13. Ratio of IOPS and throughput from 08/24/2015 to 08/28/2015.

Fig. 14 shows the ratio of read and write of storages for the 5 days. We also obtained the statistical information at the same period and calculated IOPS and throughput per 1 hour. Fig. 14(a) plots the absolute value of IOPS for 5 days. We also found that the results in Fig. 14 are similar to those in Fig. 12. It is observed IOPS increases between 8:00 and 9:00 and other peak is caused by updates of antivirus software and simple scan in all VMs. Fig. 14(b) shows throughput for the 5 days. Throughput value has been low in 10Gb FCoE network during the daytime, however it has been high during night. This reason is the daily backup from VDI storages to backup storages.



(a) IOPS



(b) Throughput

Fig. 14. IOPS and throughput from 08/24/2015 to 08/28/2015.

VI. Related Work

There are many TCP/IP traffic analysis studies that focused on DCNs [7][8][9], enterprises [5][6][24], educational systems for schools [25], backbone [4], and university [26]. Mori et al. [4] found that the Internet traffic consists mice flows (short size and short-lived) and elephant flows (large size and long-lived). Moreover, the characteristics of traffic patterns on DCNs are different from the Internet. Previous works [7][8][9] reported that mice flows comprise 70% or 80% of flows but carry only 10% of the total traffic volume. We found the similar traffic patterns on the target VDI. Next, Miller et al. [25] analyzed TCP/IP traffic at elementary, middle, and high schools. They clarified the distribution of flow sizes and per-flow throughput. Approximately 80% flows are smaller than 10KB, and the network usage of daytime is similar to our VDI pattern. Lee et al. [26] showed the increased traffic volume in UDP on campus networks. In particular, large flows in TCP/UDP whose sizes are larger than 100KB occupy roughly 90% of total bytes. The traffic patterns are very similar to our results. Meanwhile, Wang et al. [24] analyzed VDI workload on a telecom operator. The types of applications are very similar to those of the target VDI. Moreover, Ruan et al. [6] analyzed network traffic in their TCP/IP network to be connected 1500 PCs and IP phones. They observed that most traffic goes through the core switch and its bandwidth utilization is low and showed that email and web traffic consumed more than 74% of the bandwidth usage. Because the similar applications are executed, it would be the cause of low throughput on our target VDI.

There are a few studies to investigate the VDI performance [10][11][27]. Shamma et al. [10] analyzed the IOPS and throughput of remote storage in their VDI system for office work. Their analysis results showed that the traffic is write-heavy in IOPS while it is read-heavy in throughput. They made clear it that the IOPS of Firefox is large. Although the similar applications on VDI are executed for office work, their results were different from our results. The ratio of read exchanges is large at both IOPS and throughput in our results. Park et al. [11] showed analysis results of FC traffic on VDI. However, their target VDI workload is based on a benchmark program only, and its characteristics, such as exchange size and R/W ratio, are different from our results. Pawel et al. [27] described the problems on deploying and operating the VDI at university. But, there were no measurement and analysis results.

VII. Conclusion

In this paper, we introduced the analysis results of FC traffic on the commercial VDI system with office business applications. We used actual FC traffic captured at a part of VDI systems for office work. The dataset consists roughly 1.7 billion packets during office daytime. The R/W exchanges were carefully separated based on FCP header fields. In 84 million exchanges, the write exchanges were approximately 23.7% while the read exchanges were approximately 76.3%. In both read/write exchanges, approximately 50% exchanges are 4KB and 80% exchanges are smaller than 20KB. But, we found the large exchanges whose sizes are larger than 512KB. Although only 20% exchanges are larger than 20KB, the ratio in total bytes is close to 80%. The characteristic is similar to that of mice and elephant flows on DCNs even if the types of applications are different from HPC and MapReduce. Our results can imply that it will be effective to provide different network services that are aware of the exchange characteristics. Finally, we also analyzed the logs from VDI servers and storages to verify our analysis results, and found that the similar R/W proportions from the IOPS and throughput.

In future work, we intent to investigate a major cause of the 4KB exchange and fluctuation in the transfer time. Finally, we will analyze the patterns of application workloads and the FC traffic during long terms.

Acknowledgments

We are grateful to Takeshi Horie, Toshihiro Ozawa, Kenji Wakamiya, and Junji Ogawa for supporting this work.

References

[1] VMware Horizon View http://www.vmware.com/products/horizon-view.

[2] XenDesktop https://www.citrix.com/products/xendesktop/overview.html.

[3] R. W. Kembel, *Fibre Channel A Comprehensive Introduction*. Northwest Learning Assoc., 2009.

[4] T. Mori, R. Kawahara, S. Naito, and S. Goto, "On the characteristics of internet traffic variability: Spikes and elephants," *IEICE Transactions*, vol. 87-D, no. 12, pp. 2644–2653, 2004.

[5] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, and B. Tierney, "A first look at modern enterprise traffic," in *Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement (IMC '05)*. Berkeley, CA, USA: USENIX Association, 2005, pp. 15–28.

[6] Y. Ruan, N. Anerousis, M. Srivatsa, J. Xiao, R. T. Christner, L. Farrolas, and J. Short, "Measuring enterprise network usage pattern and deploying passive optical lans," in *Proceedings of the 15th IFIP/IEEE Integrated Network Management Symposium (IM 2015)*. New York, NY, USA: IEEE Communications Society, 2015, pp. 890–893.

[7] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: measurements and analysis," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '09. New York, NY, USA: ACM, 2009, pp. 202–208.

[8] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center tcp (dctcp)," in *Proceedings of the ACM SIGCOMM 2010*, ser. SIGCOMM '10. New York, NY, USA: ACM, 2010, pp. 63–74.

[9] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, "Inside the social network's (datacenter) network," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, ser. SIGCOMM '15. New York, NY, USA: ACM, 2015, pp. 123–137.

[10] M. Shamma, D. T. Meyer, J. Wires, M. Ivanova, N. C. Hutchinson, and A. Warfield, "Capo: Recapitulating storage for virtual desktops," in *Proceedings of the 9th USENIX Conference on File and Stroage Technologies*, ser. FAST'11. Berkeley, CA, USA: USENIX Association, 2011, pp. 31–45.

[11] J. Park, Y. Kim, and Y. Kim, "Analysis of vdi workload characteristics," in *Proceedings of the The Third International Conference on Digital Enterprise and Information Systems (DEIS2015)*. Wilmington, USA: SDIWC, 2015, pp. 11–17.

[12] Hyper-V https://technet.microsoft.com/library/hh831531.aspx/.

[13] Tshark https://www.wireshark.org/docs/man-pages/tshark.html.

[14] FC fields https://www.wireshark.org/docs/dfref/f/fc.html.

[15] FCP fields https://www.wireshark.org/docs/dfref/f/fcp.html.

[16] M. Tamura, K. Iizawa, M. Maeda, J. Kato, T. Kumano, Y. Nomura, and T. Ozawa, "Distributed object storage toward storage and usage of packet data in a high-speed network," in *The 16th Asia-Pacific Network Operations and Management Symposium, APNOMS 2014*. New York, NY, USA: IEEE Communications Society, Sept 2014, pp. 1–6.

[17] Fluentd http://www.fluentd.org/.

[18] MongoDB https://www.mongodb.org/.

[19] SNMP https://www.rfc-editor.org/rfc/rfc1157.txt.

[20] Windows Management Instrumentation (WMI) https://msdn.microsoft.com/en-us/library/.

[21] WMI Performance http://wutils.com/wmi/root/cimv2/win32_perfrawdata/.

[22] WMI Client http://pkgs.org/download/wmi.

[23] Storage I/O Control Technical Overview and Considerations for Deployment http://www.vmware.com/files/pdf/techpaper/VMW-vSphere41-SIOC.pdf.

[24] F. Wang, Y. Liu, B. Lei, and J. Li, "Benchmark driven virtual desktop planning: A case study from telecom operator," in *Proceedings of the 2012 International Conference on Cloud and Service Computing*, ser. CSC '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 204–211.

[25] R. Miller, W. Matthews, and C. Dovrolis, "Internet usage at elementary, middle and high schools: A first look at k-12 traffic from two us georgia counties," in *Proceedings of the 11th International Conference on Passive and Active Measurement (PAM'10)*. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 151–160.

[26] C. Lee, D. K. Lee, and S. Moon, "Unmasking the growing udp traffic in a campus network," in *Proceedings of the 13th International Conference on Passive and Active Measurement (PAM'12)*. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 1–10.

[27] C. Pawel, "Implementation of virtual desktop infrastructure in academic laboratories," in *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*. Washington, DC, USA: IEEE Computer Society, 2014, pp. 1139–1146.