Storage Offload on SmartNICs

Purna Chandra Mandal[1], Nayana Mariyappa[2], Souryendu Das[3], and Anbuvelu Venkataraman[4]

[1,2,3,4]Network Software Engineer
[1,2,3,4]Network Division Software Cloud Team, Data Center Group, Intel
[1]purna.chandra.mandal@intel.com
[2]nayana.mariyappa@intel.com
[3]souryendu.das@intel.com
[4]anbuvelu.venkataraman@intel.com

***Abstract*—Cloud Service Providers (CSPs) nowadays are servicing huge amount of network and storage traffic in their data centers imposing huge cost and burden on their server system. The problem becomes worse with the introduction of virtualization technologies which increases traffic manifolds. Applications such as software-defined storage (SDS) and big data also increase traffic between servers, and often Remote Direct Memory Access (RDMA) is used to accelerate storage data transfers between servers. In this paper we propose to virtualize the storage capabilities of the host server and offload them to a SmartNIC, reducing load on host CPU, making the system robust in a cost effective way. We also demonstrate that SmartNICs can do this virtualized networked storage in a more efficient, easier to manage and look-a-like to local physical storage.**

***Index Terms*—CSPs, Storage Offload, SmartNIC, NVMe, Custom Logic, CPU exhaustive**

I. Introduction

There is an increasing demand for storage spaces and network functionality in today's world for complex and performance sensitive applications. With the advent of virtualized system and pay-per-use business model there is an ever increasing pressure on CSPs to provide the users with the best possible performance and I/O bandwidth, with reduced latency on per request basis. These virtualized solutions employ multiple virtual machines (VMs). I/O requests coming from these VMs are processed by vHost running on Host server and transfer them to host Network and Storage controller. The host servers thus exhaust out in terms of CPU performance and memory whenever there is burst of requests. Overall network throughput gets impacted, latency increases and performance degrades.

Moreover CSPs deploy multiple processors for security and cryptograpgy which consumes good number of host CPUs available for network packet processing and storage [1]. CSPs also sometimes tend to do a lot of live migration in their host servers, where VMs migrate from one host server to another in case of an emergency or debug scenario [2]. For doing live migration there is a requirement to modify a lot of proprietary software on the host which is cumbersome and also running those softwares is CPU intensive.

To solve this problem traditionally CSPs employed more number of host servers with added CPU computing power, more memory and enlarged storage capacity. However there are limitations to the above approach. One of them is that the additional CPUs require a lot of setup and maintenance cost and physical rack space. Also scalability is a huge problem because of under-utilization of system. To improve storage performance, CSPs have employed NVMe (Non-Volatile Memory Express) controller in their host servers for storage processing. However there is a latency bottleneck here in the implementation of shared storage or storage area networking where data needs to be transferred between the host (initiator) and the NVMe-enabled storage array (target) [3].

The alternative to the above approach has been thought as offloading some of these memory intensive and CPU exhaustive processes to additional H/W or virtualize them. Using NVMe-oF (NVMe-over-fabric) solves the latency bottleneck in the host by acting as a messaging layer between the host computer and target SSDs or in a shared system network over ultra-high speed RDMAs/Fibre Channels [4][5][6].

SmartNICs are thought to be the answer here for all the problems discussed above. Multiple features related to network packet processing, storage, cryptography, live migration, etc could be offloaded to SmartNICs which could be easily attached to the Host server over PCI Express [7][8]. This would free additional CPU power in host server which can be allocated to additional virtual machines. Like networking offload with vSwitch and crypto handling, SmartNICs can have storage controllers which efficiently manage both H/w and S/W to provide high throughput with low latency and greater performance than a networked or local storage [9]. These storage controllers access storage data by deploying NVMe-oF (NVMe-over-TCP or NVME-over-RDMA) and send storage request from Host/VM to remote cloud storage. Having Storage controllers on SmartNICs reduces complexity of maintaining software and support direct data transfer between VMs and SmartNIC over PCIe. Also it allows CSPs to deploy specific cryptograpy for storage traffic or maintain local cache for improved performance and redundancy. Moreover custom logic in SmartNIC will be able to assist the Live Migration of running VMs to different Host Server.

Fig. 1 highlights the offloading of processes from Host Server's CPU to a SmartNIC's memory core. This frees CPU cycles on the host side which could be used for other CPU intensive processes. This saves a lot of revenue as the

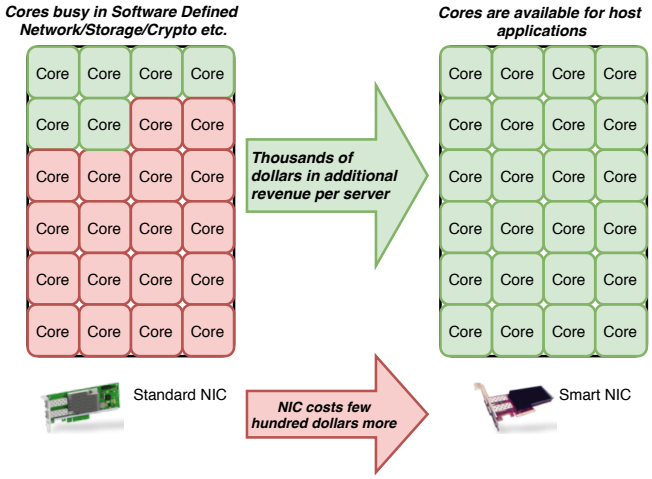

Fig. 1: Offload from Host Server to SmartNIC

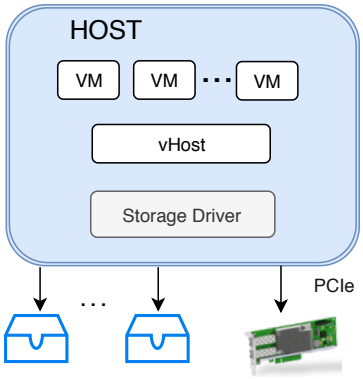

Fig. 2: Traditional Standard Setting of Storage solutions by CSPs

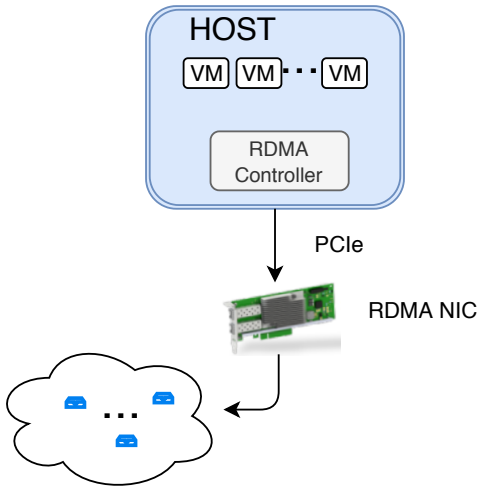

Fig. 3: Standard Setting of Storage solutions by CSPs using NVMe-oRDMA

CSP need not buy additional servers and economizes power usage (a server costs more and requires more power than a SmartNIC) [10]. Since the offload is done to a SmartNIC, it would be able to substitute many of the major functionality of Host's CPU and give added features in terms of packet processing [11].

In this paper we propose a virtualized networked storage solution which would emulate the storage capability of a host server and would free a lot of storage and packet processing memory required on the host, by offloading these activities to itself. Section II of the paper describes the system model, section III gives performance analysis for our solution and in section IV we conclude giving scope of future work.

II. System Model

A standard setting would have the host server connected to multiple storage devices. These storage devices could be physically attached like SSD or NVMe storage arrays or emulated backed by cloud storage [12]. Also there would be a standard NIC connected to the Host over which all the network and storage requests to cloud storage would be transported.

In more complex case of virtualized system, Host server would be running multiple VMs (Virtual Machine) sharing all available CPUs and memory resources. Network requests from VMs would be processed by vHost (and vSwitch) application running on Host System which in-turn transfer them to connected NIC. Similarly, storage requests would be processed by vHost application running on Host system and eventually would be terminated at local physical storage or get transferred to Cloud over NIC. Fig. 2 depicts this type of system with vHost application and local physical storage. In more advanced system, CSPs deploy NVMe or RDMA controller to commit storage requests to local storage arrays and remote cloud respectively. This improves system latency and performance. Fig. 3 depicts that type of system. The response path is exactly the opposite in both the cases. There is significant delay in these models as the request goes through multiple devices and multiple processing stages.

In our model we offload the processing of these requests (storage and network) to a SmartNIC which is connected to the host over PCI Express. Fig. 4 demonstrates our system topology and implementation. A SmartNIC has its own CPU cores and custom logic to handle control and data path for Network traffic in the most efficient way. The SmartNIC have in-built RDMA RoCEv2 controller and uses NVMe-oF internally to communicate with cloud storage over Ethernet. There are multiple storage protocols possible in a SmartNIC. CSPs can configure SPDK application running on SmartNIC to transfer the requests generated at Host either to local NVMe storage array or to NVMe-over-tcp or over NVMe-over-RDMA. CSPs can also configure cryptography algorithm for transferring storage traffic across Ethernet.

III. Performance Analysis

Fig. 5 showcases the software and hardware setup for running performance analysis test cases on the Storage Offload feature. Our SmartNIC presents itself as SR-IOV capable virtio-blk and virtio-net PFs (Physical Function). Each of the PFs and respective VFs (Virtual Functions) can be independently accessed by one of VMs directly. For performance analysis fio application running on Host server issues block read or write requests to the virtio-blk PF/VF device. SPDK

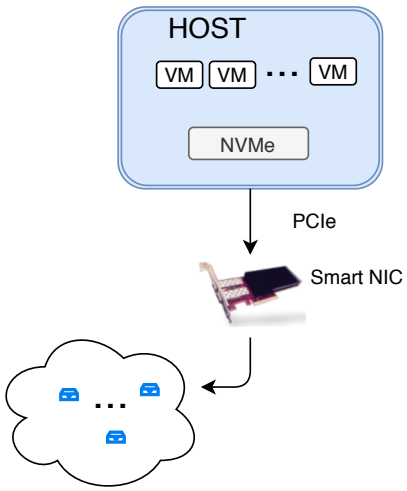

Fig. 4: Storage offload solution using SmartNIC

application running on SmartNIC gets those requests and process them. Either requests are completed at local NVMe array or over iscsi. Here we shared data with two approach – one with iscsi, intiator running on Host and storage at remote server, and other with local storage.

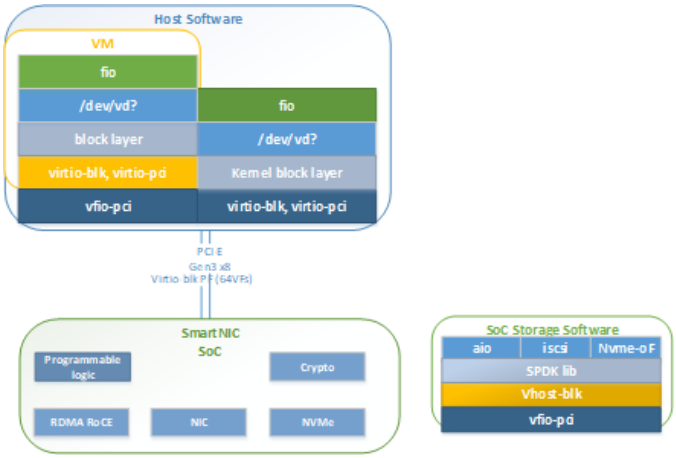

Fig. 5: Performance Analysis Setup

As per our test reports a R/W operation performed via fio with the block size of 512 bytes and queue size of 32 performed in a single virtio-blk device resulted in 1267k IOPS (Input Output Per Second). Testing with block sizes of 512 and 4096 bytes and queue sizes of 32 and 128 resulted in varied performance outputs. Table I reflects the performance test results. In 4096 bytes block size Host is able to utilize PCI Express bandwidth fully.

TABLE I: Storage Offload Performance Testing Results with SPDK

Block Size	Queue Size	IOPS
512	32	1267k
512	128	2314k
4096	32	896k
4096	128	1547k

TABLE II: Performance and latency for Host accessing cloud storage over iSCSI

IOPS	Bandwidth(Gbps)	Latency(ms)	
		P50	P99
1318k	40.24	38.14	48.89
1274k	38.8	36.6	52.99
1771k	54.04	26.75	34.04
1711k	52.24	18.816	40.704

Table II shows performance testing results on an iSCSI target device with block data size of 4096 bytes and queue size as 128. The delay in milliseconds for 50% (P50) and 99% (P99) completion of storage request is also showcased in the picture. This delay is not that significant as compared to the offloading benefits provided by SmartNIC.

IV. Conclusion

In this paper we discussed about the performance and scalability issues the CSPs face in their Servers to match with ever evolving demand of data-rich applications, in terms of I/O bandwidth, I/O latency, CPU computing power and cost. They also face in-efficiency implementing cryptography in general purpose processors and supporting live migration. SmartNICs are those devices which would help CSPs to solve the above problems as they could offload functionality like network packet processing, storage, cryptography, live-migrations to specially built hardware and software combinations. Thereby reducing cost, scalability bottleneck and also introducing abstraction. This abstraction is even more beneficial in the long run as advancement in technologies or evolving computational requirements in hardware and software would not require change in core computing system of Servers.

Specifically, on storage offloads to SmartNIC, we showed that Host can utilize maximum PCI-E bandwidth easily while processing storage requests through SmartNIC. Although there is minimal increase in latency, intelligent hardware-software design on SmartNIC can better the number. However there are still a lot of open areas which could be addressed. Our solution does not incorporate inline-cryptography, local storage caching or live-migrations in the SmartNIC. In future work, we would like to address these aspects as well.

References

[1] Mazhar Ali, Samee U. Khan, and Athanasios V. Vasilakos, "Security in cloud computing: Opportunities and challenges", *Information Sciences*, Vol. 305, pp. 357–383, 2015.

[2] Massimo Ficco, Christian Esposito, Henry Chang, and Kim-Kwang Raymond Choo, "Live Migration in Emerging Cloud Paradigms", *IEEE Cloud Computing*, Vol. 3, issue 2, pp.12–19, 2016.

[3] S. Nangare, "NVMe over Fabrics: Fibre Channel vs. RDMA", *Network Computing*, 2018.

[4] Zhengyu Yang, Morteza Hoseinzadeh, Ping Wong, John Artoux, Clay Mayers, David Thomas Evans, Rory Thomas Bolt, Janki Bhimani, Ningfang Mi, and Steven Swanson, "H-NVMe: A hybrid framework of NVMe-based storage system in cloud computing environment", *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*, 2017.

[5] Nusrat Sharmin Islam, Md. Wasi-ur-Rahman, Xiaoyi Lu, and Dhabaleswar K. Panda, "High Performance Design for HDFS with Byte-Addressability of NVM and RDMA", *Proceedings of the 2016 International Conference on Supercomputing A*, rticle No. 8, 14 pages, 2016.

[6] Zvika Guz, Harry (Huan) Li, Anahita Shayesteh, and Vijay Balakrishnan, "NVMe-over-fabrics performance characterization and the path to low-overhead flash disaggregation", *Proceedings of the 10th ACM International Systems and Storage Conference*, Article No. 16, 9 pages, 2017.

[7] Daniel Firestone, Andrew Putnam, Sambhrama Mundkur, Derek Chiou, Alireza Dabagh, Mike Andrewartha, Hari Angepat, Vivek Bhanu, Adrian Caulfield, Eric Chung, Harish Kumar Chandrappa, Somesh Chaturmohta, Matt Humphrey, Jack Lavier, Norman Lam, Fengfen Liu, Kalin Ovtcharov, Jitu Padhye, Gautham Popuri, Shachar Raindel, Tejas Sapre, Mark Shaw, Gabriel Silva, Madhan Sivakumar, Nisheeth Srivastava, Anshuman Verma, Qasim Zuhair, Deepak Bansal, Doug Burger, Kushagra Vaid, David A. Maltz, and Albert Greenberg, "Azure Accelerated Networking: SmartNICs in the Public Cloud", *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI '18)*, pp. 51–64, 2018.

[8] Ondřej Hlavatý, "Network Interface Controller Offloading in Linux", *Department of Distributed and Dependable Systems, Charles University*, 2018.

[9] Raffaele Bolla, Roberto Bruschi, Franco Davoli, and Flavio Cucchietti, "Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures", *IEEE Communications Surveys & Tutorials*, Vol. 13, Issue 2, pp. 223–244, 2011.

[10] Dexiang Wang, Janise McNair, and Alan George, "A Smart-NIC-Based Power-Proxy Solution for Reduced Power Consumption during Instant Messaging", *2010 IEEE Green Technologies Conference*, 2010.

[11] John W. Lockwood, Adwait Gupte, Nishit Mehta, Michaela Blott, Tom English, and Kees Vissers, "A Low-Latency Library in FPGA Hardware for High-Frequency Trading (HFT)", *2012 IEEE 20th Annual Symposium on High-Performance Interconnects*, 2012.

[12] Muhammad Raghib Hussain, Vishal Murgai, Manojkumar PANICKER, Faisal Masood, Brian FOLSOM, and Richard Eugene Kessler, "Systems and methods for enabling access to extensible storage devices over a network as local storage via NVME controller", *United States Patent US9294567B2*, 2016.