# Secured Large Scale Shared Storage System

Miguel Rodel Felipe, Sundaram Sivaraman, Wang Donghong, and Khin Mi Mi Aung
Data Center Technologies Division, Data Storage Institute, A*STAR, Singapore 138932
Email: {Rodel_FM, Sivaraman_S, WANG_Donghong, Mi_Mi_AUNG}@dsi.a-star.edu.sg

***Abstract*** —The integration of next generation NVM into storage network architecture requires the redesigning of current storage security infrastructure to full realize the high performance of future storage system. As a minimum, the following three layers of security control need to re-architecture: credential management at software layer secured access controls at system layer, and secured ultra-bandwidth peripheral communication at hardware layer. The main purpose of this paper is to redesign and develop storage security infrastructure to support the future storage network architecture that is based on NVM technologies.

***Index Terms***—Storage system, hardware security modules, system security

## I. Introduction

In data-driven age, many organizations face the challenge of implementing data protection and security system as the threat can significantly disrupt and damage their enterprise. With the increasing importance and emphasis on security in ever growing enterprise storage network, the proposed secured large scale shared storage system will be directly beneficial to industries and research organizations. We believe that our evaluation of the methodologies will elicit several important issues that could spur further interest from industries.

There is a need for storage system architecture design change for future data center when many believed it has reached its physical limitations. As and always, security has taken a front seat in the design change of systems and there are substantiated impacts on data security with the changes. In this paper, we address security controls in three layers of protection, such in hardware, system and software layers.

One of the important issues in multi-tenant data centres is the access control, which focuses on the protection of information against unauthorized access. As systems grow in size and complexity, access control is a special concern for distributed systems. The integration of next generation NVM technologies enable storage system offers high performance advantage and scalability. To match with highly intensive I/Os in the system, the secured access control is needed.

Manuscript received October 23, 2015; revised December 28, 2015.
This work was supported mainly by the Singapore A*STAR TSRP project 1122804006.
Corresponding author email: Rodel_FM@dsi.a-star.edu.sg
doi:10.12720/jcm.11.1.93-99

The integration next generation NVM into storage network architecture induces changes to the storage security design, specifically in the following areas:

- Security controls, such as, identity and credential management at software layer,
- Secured access controls at system layer
- Secured ultra-bandwidth peripheral communication at hardware layer,

These changes are necessary in order for the security mechanisms to match with the intensive high I/O throughput performance of information storage system.

Our innovative data protection and security mechanism for large scaled storage system will not only develop human capitals that create knowledge and technology of economic impact, but also know-how to integrate technology from many disciplines, enable technology transfer to industry.

We design a transparent and vendor-neutral approach, thus, the execution of this project can immediately establish closed collaborations not only between local institutions and industries already present in Singapore but also contribute to standard bodies and global community of knowledge generation and innovation.

The scalable approach with organization, operation and technology aspects can be deployed in service based sectors such as government, healthcare, education, retail, transportation, manufacturing and businesses in general and keyed to addressing industry needs, which may span multiple disciplines.

The rest of this paper is organized as follows: Section II gives a review of the related work. Section III introduces the proposed security architecture for future data center. Section IV describes the performance analysis of the proposed framework. Finally concluding remarks are made in Section V.

## II. Back Ground and Related Work

In memory computing needs to address several issues to achieve mainstream adoption, such as a lack of standards, the scarcity of skills, relative architectural complexity, security concerns, and monitoring and management challenges [1].

In-Memory Databases does not deliver the range of security countermeasures present in conventional databases; this includes, label based access control, data redaction capabilities, applying patches while database is online and policy management tools [2].

Despite half a century worth of research on software safety, memory errors are still one of the primary threats to the security of our systems. Common Vulnerabilities and Exposures (CVE) show~20% of vulnerability reports are due to memory errors and 63% of cases from the most popular exploit toolkits are focused on exploiting memory vulnerabilities [3]. One area of research to reduce the potential damage caused by exploitable vulnerabilities is containment mechanisms.

In direct persistent memory access [4], memory protection is through existing methodologies, it means by virtual address space and page faults). It might not be sufficient for persistent memory data as data resides beyond a process' execution lifetime. Existing memory protection methodologies are more suitable for ephemeral data.

In virtual file system access controls [5], existing vfs provides discretionary and mandatory (e.g SELinux) access controls to persistent files and it is sufficient if NVM is considered only as a file repository. However, it requires an additional process dimension, to allow only processes creating NVM data to access them.

The data being produced is increasingly unstructured, complex to manage and require different granularity of security. The rapid growth of data will impose challenges to the information storage systems in data centers.

Ceph is a distributed storage system and the protection offered by Ceph authentication is between the Ceph client and the Ceph server hosts. The limitation of Ceph is that the cephx protocol authenticates Ceph clients and servers to each other. It is not intended to handle authentication of human users or application programs run on their behalf [6].

## III. Security Architecture for Future Data Centres

This section will tackle the security controls that the authors propose on different layers of the system when deploying NVM-based data centers. This security architecture is mainly driven by the increase in the efficiency and performance of future data centers. There is a need for new security controls on the storage, operating system, and application layers when adopting NVM-based systems because the existing security controls are mainly designed to adopt to the fast and volatile characteristics of the primary memory and to the slow and persistent characteristics of the secondary memory. Existing security solutions on the storage layer like full-disk encryption and file-system encryption are not designed for performance because the storage layer (e.g. HDD) is slow anyway, thus, software encryption solutions are generally acceptable. File-systems are generally designed to compensate for the slow nature of secondary storage, thus, having to use the same technique for NVM-based systems, will incur a lot of overhead. Providing file-system encryption will add more overhead to this already inefficient technique. This security architecture will alleviate this overhead by introducing security mechanisms that fit NVM-based systems well.

Fig. 1 shows the target areas of protection in the host system. The following sections discuss the security architecture proposal of this paper.

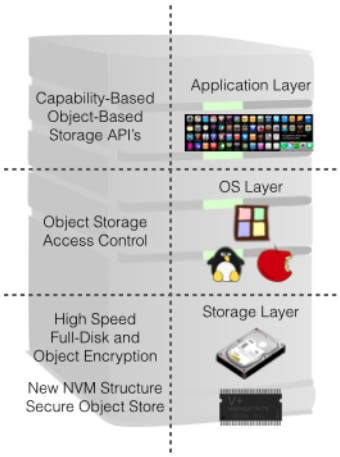


Fig. 1. Security controls for secure large scale shared storage system.

### A. Security in the Storage/Hardware Layer

NVM can be deployed as a primary storage or a secondary storage. With the various possibilities of the use-cases for NVM, there should be an efficient way to access this new form of storage. The current protection mechanisms for DRAM, paging, is not sufficient because of it is designed with volatile nature of DRAMs. Various methods exist like to keep the page descriptor entries intact when the system reboots. However, with the advancement of NVMs such increasing capacities and byte-addressability, this solution might not be efficient. As a secondary storage, some solutions like provide a new file system on top of the NVM, which also uses the existing memory protection techniques, i.e. paging.

Because of the characteristics of the NVM (persistence, capacity, and speed), the authors propose an object-based approach in accessing NVMs using the Secure Object Stores (SOS) [7]. This solution provides a protection mechanism in object or domain granularity. At the hardware level, the proposition is to organize the regions of the NVM into owners, domains, and objects. Object region is the main data repository containing named memory objects created / accessible to programmers. Domains are containers for objects that are used to provide memory protection boundaries to processes during runtime and also act as access control boundaries. The domain region in NVM stores domain related meta-data such as domain names, objects under domains etc. Owners are system users who own memory domains, all memory domains created by applications are by default owned by the system user under which the application is run. Further, applications that create domains / objects by default are provided with access capabilities.

NVM performance has reached speeds such that it has been considered as a primary memory replacement.

Providing an encryption methodology that can support DRAM speeds is currently non-existent. Encryption is one of the most resource-hungry processes in a system, so offloading this process into a separate hardware is the most efficient way to accelerate encryption. However, existing Hardware Security Modules (HSM) do not provide symmetric encryption performance that is required for the servers in the next generation data centres. Using these HSMs will make the encryption as the bottleneck of the data centre's performance.

A new technology that optimizes the way the HSM is controlled and where the encryption process is pipelined is proposed as the primary encryption processor for the future data centres. This technology is called hardware security accelerator (HSA) and this technology provides a 7x symmetric encryption improvement from one of the market leader's HSMs, Safenet Luna PCIe. [8]. HSA is currently developed using Virtex 6 FPGA development board with PCIe Gen. 1 interface. When the firmware is ported into a faster interface (e.g. PCIe Gen. 3), the performance of HSA will improve dramatically, which will make encryption an integral part of the future data centres as NVM-based systems continue to be developed.

### B. Security *in the* Operating System Layer

We proposed an object-based storage mechanism when accessing the NVM. One of the reasons why the authors chose object stores is that it provides interfaces that make programming with persistent data easier. It also provides high-level abstractions, which makes it easy to share objects across applications. Object stores exports a common interface on which objects can be shared across different platforms. The most important aspect of object stores is that security and privacy mechanisms can be applied at object granularity.

The use of capability-based security mechanism for the object stores is proposed in this paper. The following lists the advantages of capability-based systems:

- It allows persistent objects to be referred through tokens (similar to virtual address) in programs, which provides memory protection boundaries to processes accessing them.
- It allows us to view persistent memory uniformly across applications; therefore, data can be effectively shared by sharing capability tokens.
- Persisting capability tokens allows applications to access data that it stored on NVM directly, across application and system restarts.
- It encodes access control information.

In the OS level, SOS provides a Local Node Security Gateway (LNSG) that provides the security mechanisms of the system, i.e. access control and cryptographic operations. It provides application API's that allow storing, reading, modifying, and sharing objects within the host system. The LNSG contains four key components, i.e. the key manager, the capability manager, the capability generator and the cipher engine. These components manage the NVM regions as well as provide capability based access control to the application layer.

### C. Security in the Application Layer

Cloud object stores like Amazon S3 and Openstack's Swift object storage services have gained a very high adoption rate from Enterprises and Small-to-Medium Businesses (SMB) in recent years and are expected to grow further. IDC forecast that worldwide revenue for file-based and object-based storage would reach $38 billion by 2017.

Applications using the API's for object stores have various authentication/authorization mechanisms based on the various object services. In this paper, the authors are proposing to extend the capability-based object storage (SOS), currently limited within a single host/server, into a large-scale shared object storage system.

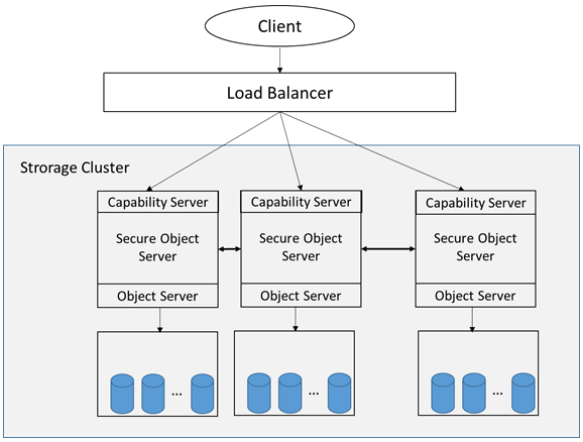


Fig. 2. Distributed SOS architecture

Capability Generator on the LNSG encodes and encrypts capability tokens upon request by the capability manager. Each capability token for a domain and/or object encodes object/domain identifiers, object location information and access rights. In order to expand the scope of access control from local node to a larger-scale, the Capability Generator has to be designed such that it follows the architecture of the shared storage system. If the object storage system is centralized, e.g. SwCift, there has to be a central Capability Generator (within the Proxy server in the case of Swift) controlling the Capability Generator Agents in the LNSG (replacing the Capability Generator). If the object storage system is distributed, e.g. Ceph, the Capability Generator of each host should form a cluster. These techniques will ensure that the capability tokens will cover the scope of the large-scale object storage service.

The design of the SOS integrates seamlessly with services that offer object storage systems like Amazon's S3 and Swift. The subscribers can be treated as the owners; the application that uses the object storage service can be treated as a domain. The capability tokens that the application holds provide the access control that it has on an object stored on the storage service.

### D. Distributed and Centralized SOS Architecture

In a distributed architecture Capability generator/Server is maintained inside the object servers in a Cluster. These servers communicate with each other to generate/authenticate and manage Capability tokens.

In a Centralized SOS Architecture, as briefly explained in the previous section we propose to have the Capability Generator as a separate entity central to the Architecture. A diagrammatic representation is given below:

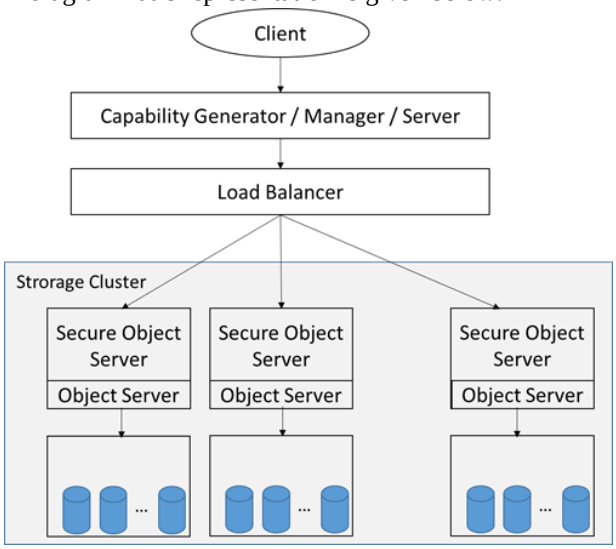


Fig. 3. Centralized SOS architecture

## IV. Experimental Results

The authors integrated HSA in a petabyte-scale storage system to see if the performance of the cryptographic processing system is within the target limit of 5% overhead. The HSA was used as the cryptographic engine in doing the encryption of the disk partitions used as the metadata and data partitions of the Hybrid File Systems (HFS) [9]. The HFS is the primary file-system used by the object storage nodes of the Ceph Storage. The PB-scale storage system also optimizes the Ceph Metadata cluster using the DROP metadata management system [10].

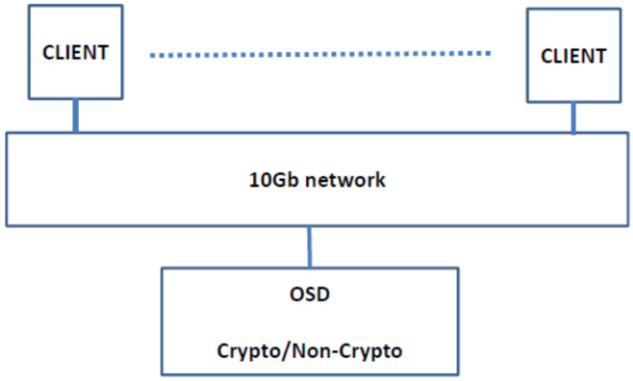


Fig. 4. Ceph storage cluster topology for Crpyto/Non-Crypsto OSD node integration

The HSA is deployed on a PB-scale storage system to measure its performance when doing reads and writes using small (4KB) and large (4MB) IO sizes. The cryptographic overhead for small reads/writes and large reads is below the target performance of 5%. The cryptographic overhead of large writes is very large (~15%) and it is due to some of the kernel's optimization process (e.g. caching) that report finished writes even if the data is not yet written on the actual storage media. The cryptographic layer does not provide the same optimization mechanisms.

For the performance tests, a ceph storage cluster is set up as shown in Fig. 4. The hardware configuration of the test machines is shown in Fig. 5.

| node | # of nodes | CPU | Memory |
|---|---|---|---|
| Client | 4 | Intel(R) Xeon(R) CPU X3220@2.40GHz, 4 cores | 4GB |
| OSD | 1 | Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz, 8 cores | 12GB |

| HDD | SSD storage | |
|---|---|---|
| 1 x 1TB, 7200rpm | 0 | |
| OSD DATA | OSD Journal | HFS metadata |
| 100GB, 7200rpm | 1GB | 20GB |

Fig. 5. Ceph storage cluster hardware configurations

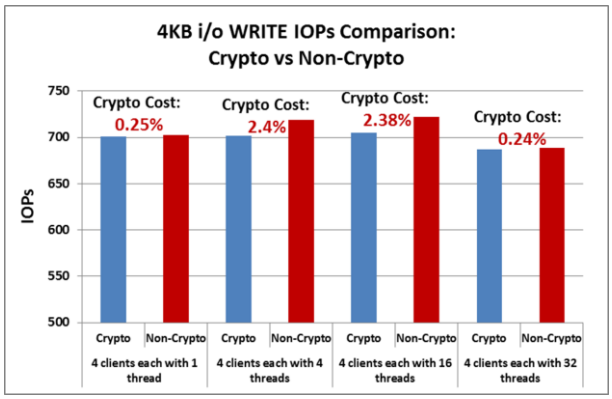


Fig. 6. 4KB i/o WRITE IOPs comparison: Crypto vs Non-Crypto.

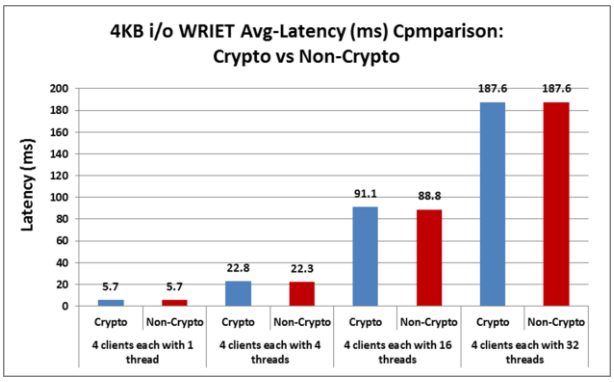


Fig. 7. 4KB i/o Write Avg-Latency (ms) comparison: Crypto vs Non-Crypto.

In small I/O size random write tests, 4KB size I/O random write workload is used. Fig. 6 and Fig. 7 show the performance in IOPs and latency comparison between with DSI cryptographic and without cryptographic. There is less than 2.4% overhead introduced by DSI cryptographic technology.

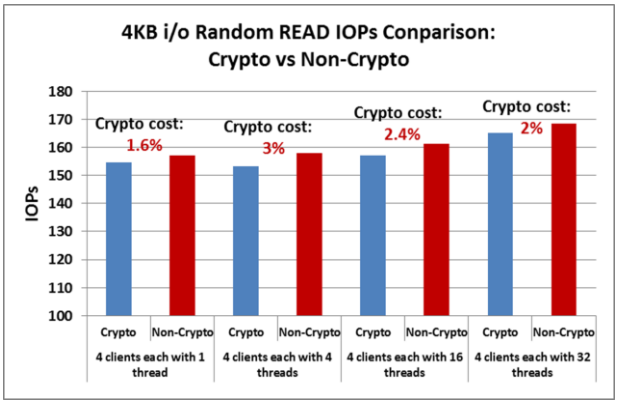


Fig. 8. 4KB i/o Random READ IOPs comparison: Crypto vs Non-Crypto.

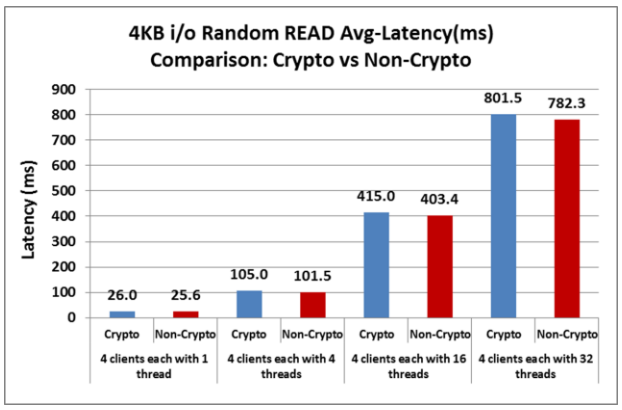


Fig. 9. 4KB i/o Random READ Avg-Latency (ms) comparison: Crypto vs Non-Crypto.

In small I/O size random read tests, 4KB size I/O random read workload is used. Fig. 8 and Fig. 9 show the performance in IOPs and latency comparison between with DSI cryptographic and without cryptographic. There is less than 3% overhead introduced by DSI cryptographic technology.

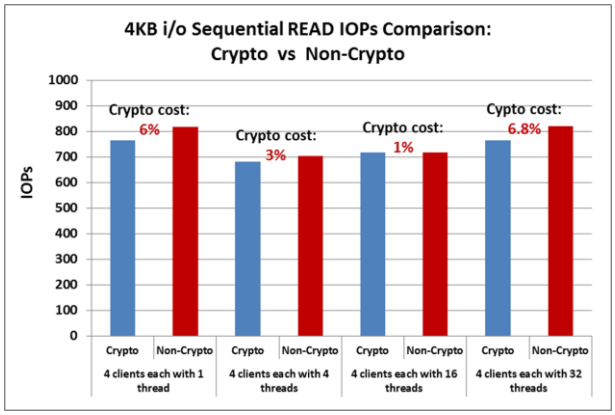


Fig. 10. 4KB i/o Sequential READ IOPs comparison: Crypto vs Non-Crypto.

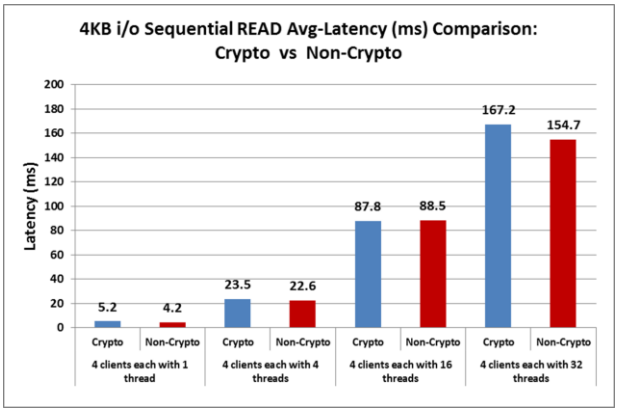


Fig. 11. 4KB i/o Sequential READ Avg-Latency(ms) comparison: Crypto vs Non-Crypto.

In small I/O size sequential read tests, 4KB size I/O sequential read workload is used. Fig. 10 and Fig. 11 show the performance in IOPs and latency comparison between with DSI cryptographic and without cryptographic. There is less than 6.8% overhead introduced by DSI cryptographic technology.

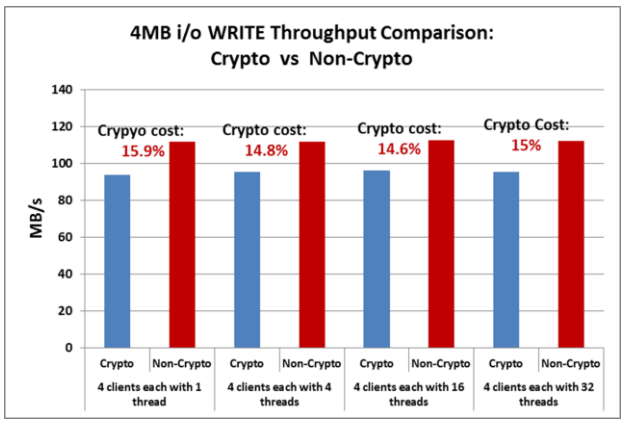


Fig. 12. 4MB i/o WRITE Throughput Comparison: Crypto vs Non-Crypto.

In big I/O size random write tests, 4MB size I/O random write workload is used. Fig. 12 shows the throughput in MB/s comparison between with DSI cryptographic and without cryptographic. There is less than 15.9% overhead introduced by DSI cryptographic technology.

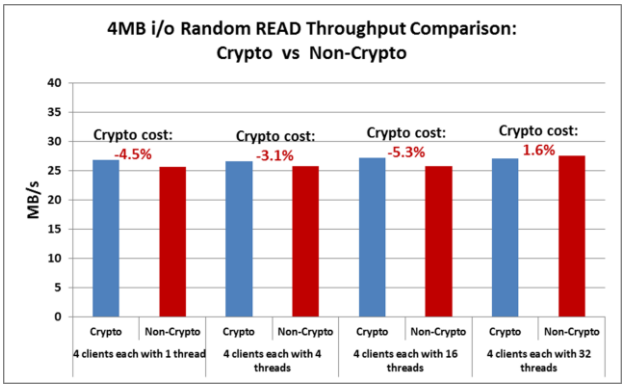


Fig. 13. 4MB i/o Random READ throughput comparison: Crypto vs Non-Crypto.

In big I/O size random read tests, 4MB size I/O random read workload is used. Fig. 13 shows the throughput in MB/s comparison between with DSI cryptographic and without cryptographic. There is less than 1.6% overhead introduced by DSI cryptographic technology.

In big I/O size sequential read tests, 4MB size I/O sequential read workload is used. Fig. 14 shows the throughput in MB/s comparison between with DSI cryptographic and without cryptographic. There is less than 0.4% overhead introduced by DSI cryptographic technology.

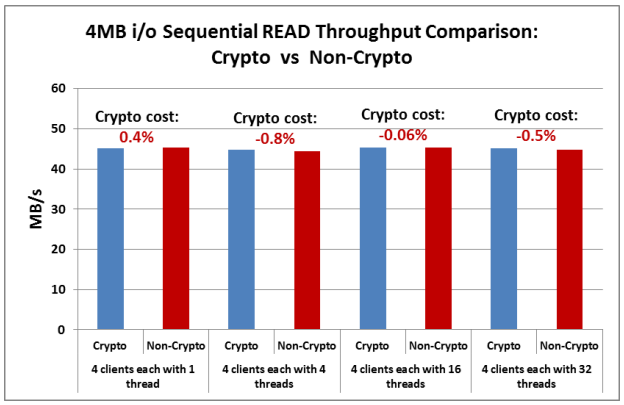


Fig. 14. 4MB i/o Sequential READ throughput comparison: Crypto vs Non-Crypto.

## V. Conclusions

In this paper, we have discussed that the NVM-based systems will be the main component of future data centres. The security infrastructure should focus on the characteristics of NVM-based systems such as persistence, capacity, and speed. This work proposes to use a high-speed cryptographic processing hardware to be used for encrypting contents of the primary and/or secondary storage. This ensures that security will not be the bottleneck of the storage system. This work also proposes a way to manage the NVM on a local host where the security mechanisms can be extended to a large-scale storage system. We also have proven that the hardware security mechanism in this paper performs well on a high-performance shared storage system.

## Acknowledgment

This work was supported mainly by the Singapore A*STAR TSRP project 1122804006.

## References

[1] Gartner says in-memory computing is racing towards mainstream adoption. [Online]. Available: http://www.gartner.com/newsroom/id/2405315

[2] [Online]. Available: http://layersevensecurity.com/blog/2013/10/31/security-in-sap-hana-the-challenges-of-in-memory-computing-2

[3] V. D. Veen, Victor, L. Cavallaro, and H. Bos, "Memory errors: the past, the present, and the future," *Research in Attacks, Intrusions, and Defenses. Springer Berlin Heidelberg*, 2012, pp. 86-106.

[4] M. E. Aho, T. M. Gooding, M. B. Mundy, A. T. Tauferner, "Remote direct memory access ('RDMA') in a parallel computer", *US Patent number* 8874681, 2014. [Online]. Available: https:// www.google.com.sg/patents/US8874681

[5] M. Abd-El-Malek, M. Wachs, J. Cipar, K. Sanghi, G. R. Ganger, G. A. Gibson, M. K. Reiter, "File system virtual appliances: Portable file system implementations", *ACM Transactions on Storage*, 2012, vol. 8, no. 3, pp. 9.

[6] M. J. Brim, D. A. Dillow, S. Oral, B. W. Settlemyer and F. Wang, "Asynchronous Object Storage with QoS for Scientific and Commercial Big Data", in *Proc. ACM 8th Parallel Data Storage Workshop*, 2013, pp. 7-13.

[7] R. F. Miguel, R. Shuqin, and K. M. M. Aung. Cipher Devices and Cipher Methods, WO 2014062136 A1. [Online]. Available: https://www.google.com/patents/WO2014062136A1

[8] Q. Q. Xu, *et al*., "DROP: Facilitating distributed metadata management in EB-scale storage systems," in *Proc. IEEE 29th Symposium on Mass Storage Systems and Technologies*, 2013, pp. 1-10.

[9] J. Chao, *et al*., "HiSMRfs: A high performance file system for shingled storage array," in *Proc. IEEE 30th Symposium on Mass Storage Systems and Technologies*, 2014.

[10] R. F. Miguel, S. Sundaram, and K. M. M. Aung, "Secure Object Stores (SOS): Non-Volatile Memory Architecture for Secure Computing," in *Proc. 4th International Conference on Communication and Network Security*, 2014.

**Rodel Felipe Miguel** received his B.Sc degree in Computer Engineering (cum laude) from AMA Computer College, Makati City, Philippines. He has more than 10 years of software development experience focused on embedded systems, device drivers, and network applications. He is currently a senior research engineer in Data Storage Institute, Singapore. His research interests include Storage and Network Security.

**Sundaram Sivaraman** received his B.Sc degree in Engineering (1st Class Honors) from Anna University Chennai and M.Sc degree from Nanyang Technological University, Singapore. He is currently a research engineer in Data Storage Institute, Singapore. His research interests include Storage and Network Security.

**Wang Donghong** received her B.E. degree major in Telecommunications from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China and M.Sc. degree major in Computer Engineering from the National University of Singapore (NUS), Singapore. She is currently working as a senior research engineer in Data Storage Institute, Singapore. Her research interests include parallel and distributed data storage system, file system and operating system, NVM storage system, Linux kernel development, and simulation tool implement for NVM storage system.