Manchester Metropolitan University

e-space

Manchester Metropolitan University's Research Repository

Please cite the Published Version

Qureshi, NMF, Siddiqui, IF, Abbas, A, Bashir, AK, Nam, CS, Chowdhry, BS and Uqaili, MA (2020) Stream-Based Authentication Strategy Using IoT Sensor Data in Multi-homing Sub-aqueous Big Data Network. Wireless Personal Communications: an international journal, 116 (2). pp. 1217-1229. ISSN 0929-6212

DOI: https://doi.org/10.1007/s11277-020-07215-3

Publisher: Springer Verlag

Version: Accepted Version

Downloaded from: https://e-space.mmu.ac.uk/627622/

Usage rights: © In Copyright

Additional Information: This is an Author Accepted Manuscript of an article published in Wireless Personal Communications: an international journal.

Enquiries:
If you have questions about this document, contact openresearch@mmu.ac.uk. Please include the URL of the record in e-space. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from https://www.mmu.ac.uk/library/using-the-library/policies-and-guidelines)

Wireless Personal Communications manuscript No.
(will be inserted by the editor)

Stream-based Authentication Strategy using IoT Sensor Data in Multi-Homing Sub-aqueous Big Data Network

Nawab Muhammad Faseeh Qureshi · Isma Farah Siddiqui · Asad Abbas · Ali Kashif Bashir · Bhawani Shankar Chowdhry · Muhammad Aslam Uqaili

Received: date / Accepted: date

Abstract Big data analytics has addressed many in-place and remote network issues in a sub-aqueous distributed computing environment. Recently, a new phenomenon is introduced in the data analytics clusters that focus on multi-homing network connectivity procedures among off-ground multiple nodes of the large-scale on-running wireless industrial applications. In this way, the clusters perform multi-layer cross-connected task processing among various networks simultaneously and perform stream based data block placement over multiple nodes in a sequential order. This satisfies the procedural performance of the cluster; however, security remains an open issue in it because

N.M.F. Qureshi
Department of Computer Education, Sungkyunkwan University, Seoul, South Korea
E-mail: faseeh@skku.edu

I.F. Siddiqui
Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro, Pakistan
E-mail: isma.farah@faculty.muet.edu.pk

Asad Abbas
Department of Software Engineering, University of Lahore, Pakistan
E-mail: asad.abbas@se.uol.edu.pk

A.K. Bashir
School of Computing, Mathematics and Digital Technology, Manchester Metropolitan University, United Kingdom
E-mail: dr.alikashif.b@ieee.org

B.S. Chowdhry
Faculty of Electrical, Electronics, and Computer Engineering, Mehran University of Engineeringand Technology, Jamshoro, Pakistan
E-mail: bhawani.chowdhry@faculty.muet.edu.pk

M.A. Uqaili
Department of Electrical Engineering, Mehran University of Engineering and Technology, Jamshoro, Pakistan
E-mail: vc@admin.muet.edu.pk

of unavailability of inter-network data block processing authorization. In this paper, we propose a stream based authentication mechanism, that specifically addresses security concerns of multi-homing sub-aqueous big data networks and presents a key authorization infrastructure that performs a proper handing taking among multiple off-ground Datanodes before an inter-network data block exchange. The simulation results depict that our approach increases multi-homing network compatibility and reliability while processing a data block in the sub-aqueous distributed computing environment.

Keywords Big data · Multi-homing wireless networks · sub-aqueous IoT sensor · intranet security · Inter-network authentication.

1 Introduction

Big data analytics is a phenomenon that performs large-scaled dataset processing in a distributed computing environment [1]. This processing is carried out through some system software applications such as Cloudera [2], Apache Hadoop [3] and MapR [4]. Apache Hadoop is widely used due to its open-source nature and flexibility of being deployed on the low configuration cluster with four components i.e. YARN, MapReduce, HDFS and Hadoop commons. YARN is a controller, that distributes the resources such as memory and computing capacity evenly on the network nodes [5]. MapReduce is a programming model that fairly performs the task to distributed nodes and process large-scale datasets on the cluster [6]. Hadoop Distributed File System (HDFS) is the cluster file system that systematically manages datasets and provide read, write and execute function on the cluster and Hadoop commons are the useful libraries to execute all these daemon threats throughout the cluster [7].

Recently, it is observed that Apache Hadoop is connecting multiple clients having different file systems beside HDFS through sub-aqueous multi-homing network that simplify the usage of various network application executions at the same time and for that purpose, it requires a separate authorization mechanism. We find several approaches such as Kerberos [8], network-layer authorization protocols [9] and KEAP [10,11] that address single network authentication support, however, we do not find any scheme that processes various networks clients through a single authorization procedure as shown in Figure-1. Moreover, due to the underwater dataset processing, the data chunks are observed with a ratio of low communication bandwidth in the central repository [12].

To solve this issue, we propose Multi-Homing Key Exchange Authorization Protocol (MEAP) that enables NFS (Network File System) to connect multiple data nodes of various networks with a security and reliable way of communication. The MEAP-enabled client decreases intra-network connectivity overhead and provides a trusted way among nodes to communicate with each other through a secure umbrella. In this way, we obtain a substitute not to use the low communication bandwidth but, acquire a trusted intra-network

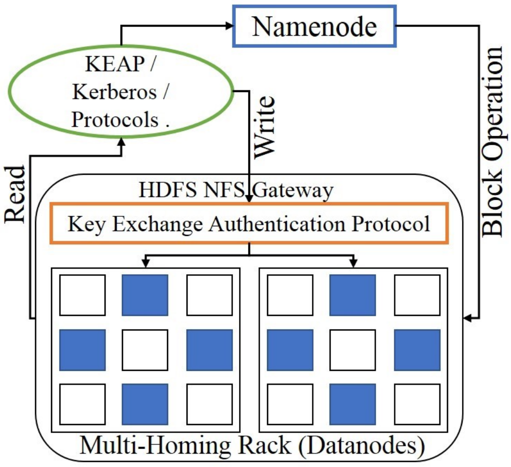

Namenode
KEAP / Kerberos / Protocols .
Write
Block Operation
Read
HDFS NFS Gateway
Key Exchange Authentication Protocol
Multi-Homing Rack (Datanodes)

Fig. 1: Default Multi-homing HDFS Cluster

node to store sub-aqueous dataset to nearby datanode.

The main contributions of the proposed scheme are:

- A novel public key encryption strategy to encode various clients in a multi-homing sub-aqueous bucket.
- A novel private key decryption strategy at HDFS to decode the trusted multi-homing sub-aqueous clients.

The remaining paper is organized as follows. Section II discusses related work. Section III briefly explains proposed approach MEAP. Section IV depicts experimental environment and evaluation result for MEAP-enabled client. Finally, section V shows conclusion and future research directions.

2 Related Work

Kerberos is a long awaiting scheme that requires a huge time to interact with network nodes, therefore, it was skipped for having a huge session lagging issue and latency timeouts [13]. The schemes related to HDFS security are divided into two types i.e. Delegation Token (DT) and Block Access Token (BAT). Both of the schemes are compatible with a mode of communication known as privileged and non-privileged clients, however, do not incorporate any mechanism to identify incoming request from a diverse network. KEAP [11] is a single network authentication protocol that authorizes network nodes to connect with each of the available datanode reliably, however, when we incorporate KEAP with multi-homing network, it fails to identify clients having contrast IPs than a single network IP.

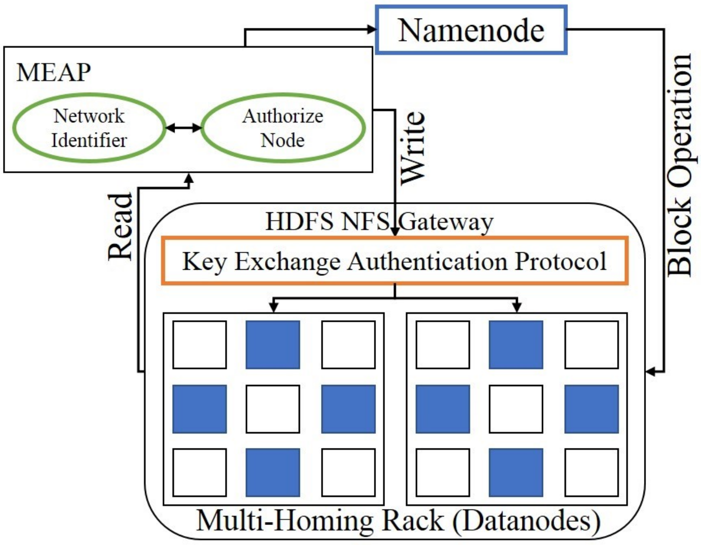


Fig. 2: MEAP-enabled Multi-homing HDFS Cluster

In order to fill this gap of requirement, we propose MEAP-NFS that enables a novel multi-homing key exchange protocol to authenticate clients from multiple networks and allows them to connect to the cluster for processing large-scale datasets in multi-homed distributed environment.

3 Multi-Homing Key Exchange Authentication Protocol

MEAP works in two stages such as (i) Public and Private key certificate generation and (ii) Application of certificates for MEAP client. At first, the scheme generates public and private key certificate and configure allowed networks to identify and exchange those certs. In the second part, HDFS allows a MEAP-enabled client to connect over its file directory and ensure trust as shown in Figure-2.

3.1 Public and Private key certificate generation

Namenode generates the public and private key certificates according to access control list (ACL) of users $user_{ACL_i}$ with $Rights_{User_i}$ and distribute to the trusted Networks $Network_{ID}$ for connecting in a reliable manner.

3.1.1 Public Key Certificate (PKC)

The steps involved in generating PKC includes product of prime integer p Diffie-Hellman group [14] of Sophie Germain prime [15], random number gen-

Table 1: MEAP Notations

Notations	Description
$user_{ACL_i}$	A user i defined in access control list
$Cert_{Pub_i}$	Public Key Certificate
$Cert_{Priv_i}$	Private Key Certificate
Pub_{Key_i}	Public Key
$Priv_{Key_i}$	Private Key
$Cert_{Req_i}$	Certificate Request
Fun_{srf}	Pseudo Random Function
C_i	Client Instance
HNG_i	HDFS NFS Gateway Instance
$Rights_{User_i}$	ACL users' rights
$Network_{ID}$	Trusted Network Information

eration g is primitive root modulo of integer p and trusted network information $Network_{ID}$ that equals to the Eulers totient function [16]. The resultant function $\Phi\left(n\right) = \left(p \times q\right)$ is swaps generated value with $Public_{Key_i}$ value and generate Certificate Authority (CA). The $Public_{Key_i}$ can be generated as:

$$Public_{Key_i} = \left(p_{int_i}, g_{mod_p}, \Phi\left(n\right), Network_{ID}\right) \tag{1}$$

Now, MEAP needs to incorporate digital signature DS_{Cert_i} [17] with public key Pub_{Key_i} to generate a certificate. The obtained certificate $Cert_{Pub_i}$ can be generates as:

$$Cert_{Pub_i} = \left(DS_{Cert_i}, Pub_{Key_i}\right) \tag{2}$$

The users $user_{ACL_i}$ of HDFS are then encrypted through generated certificate $Cert_{Pub_i}$ and generates a message. The formulated message $Message_E$ is obtained as:

$$Message_E = \left(user_{ACL_i}, Cert_{Pub_i}\right) \tag{3}$$

3.1.2 Private Key Certificate (PRKC)

The generation of private key certificate involves the same ingredients such as prime integer p and a number generator g but with inverse multiplication modular function $w = b\left(mod\phi\left(n\right)\right)$ that is extracted through a co-prime number obtain through $\phi\left(n\right) w_p = m\; mod\; \left(p-1\right)$, $w_g = w\; mod\; \left(q-1\right)$ and $g_{inv} = g^{-1}\; mod\, p$. Finally, a private key $Priv_{Key_i}$ could be generated that resolves the first step for decoding public key Pub_{Key_i} as:

$$Priv_{Key_i} = \left(p_{int_i}, g_{mod_p}, w, w_p, w_g, g_{inv}\right) \tag{4}$$

Now, MEAP needs to incorporate digital signature DS_{Cert_i} with private key $Priv_{Key_i}$ to generate a certificate. The obtained certificate $Cert_{Priv_i}$ can be obtained as:

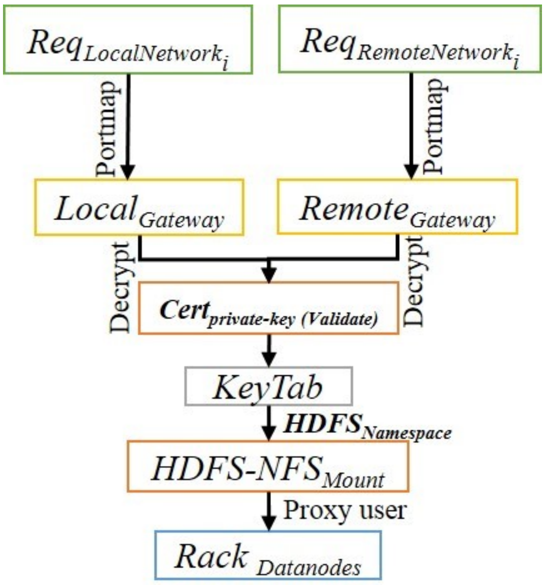


Fig. 3: MEAP-enabled Authentication mechanism of Multi-homing HDFS Cluster

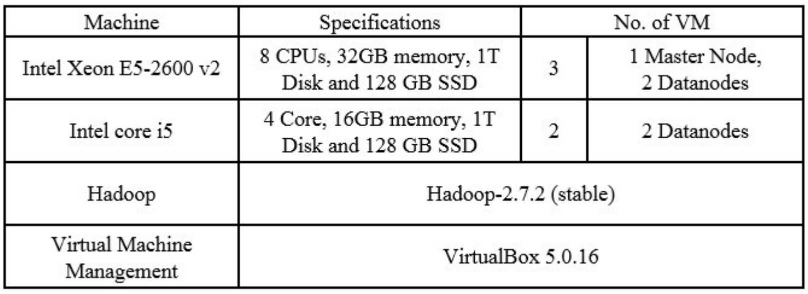
$$Cert_{priv_i} = (DS_{Cert_i}, Priv_{Key_i}) \tag{5}$$

HDFS gateway decrypts the previous message $Message_E$ that also involves a network information $Network_{ID}$ through a certificate of validation $Cert_{Priv_i}$. The final decrypted message can be decrypted as:

$$Message_D = (user_{ACL_i}, Cert_{priv_i}, Network_D) \tag{6}$$

3.2 Application of public and private certificates over MEAP client

The second phase requires identifiers to know whether the incoming connecting is coming from local network $Req_{LocalNetwork_i}$ or remote network $Req_{RemoteNetwork_i}$ through cluster portmap configuration. After that, the client information is passed through gateways such as $Local_{Gateway}$ and $Remote_{Gateway}$, the message $Message_E$ is decrypted using $client_{Session_i}$ having network $Network_{ID}$ and user information $right_{user_i}$ through *keytab*. The *keytab* is a set of principles to allocate $HDFS_{Namespace}$ and assign session $client_{Session_i}$ over HDFS directory mount point / with rack of nodes $Rack_{Datanodes}$ as illustrated in Figure-3.



Machine	Specifications	No. of VM	
Intel Xeon E5-2600 v2	8 CPUs, 32GB memory, 1T Disk and 128 GB SSD	3	1 Master Node, 2 Datanodes
Intel core i5	4 Core, 16GB memory, 1T Disk and 128 GB SSD	2	2 Datanodes
Hadoop	Hadoop-2.7.2 (stable)		
Virtual Machine Management	VirtualBox 5.0.16		

Table 2: Hadoop Cluster

Table 3: Hadoop Cluster Virtual Machines Configuration

Node	CPU	Memory	Disk	Configuration
Master Node	6	16 GB	HDD & SSD	Intel Xeon
Slave1	2	4GB	HDD & SSD	Intel Xeon
Slave2	2	4GB	HDD & SSD	Intel Core i5
Slave3	2	4GB	HDD & SSD	Intel Core i5
Slave4	2	4GB	HDD & SSD	Intel Core i5

4 EXPERIMENTAL EVALUATION

In this section, we evaluate MEAP approach over cluster configuration as seen from Table-2.

4.1 Environment

The cluster configuration consists of Intel Xeon processor with 8 CPUs, 32GB memory and storage device i.e. 1TB Hard disk drive. In addition to that, we use Intel core i5 with 4 Core, 16GB memory and storage device i.e. 1TB Hard disk drive. We install 5 virtual machines having VirtualBox 5.0.16 as seen from Table-3.

4.2 Experimental Dataset

MEAP evaluation dataset consists of: (i) 25 sub-aqueous IoT data blocks of 64MB having total length of 1.56GB sensory dataset size [18, 19].

4.3 Experimental Results

The evaluations carried out for performance measures are: (i) Local network client access and (iii) Multi-homing network client access.

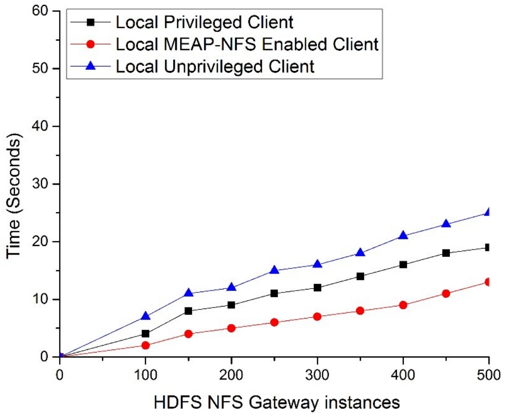


Fig. 4: Local network clients connectivity over HDFS NFS Gateway

4.3.1 Local network client access

In order to observe the local network clients connectivity, we executed lookup analysis [20] over three types of local network connections i.e. Local MEAP-NFS enabled client, Local Privileged client, and Local unprivileged client. We observed that the above-mentioned clients consist of 500 HDFS NFS gateway instances and evaluated that Local MEAP-NFS enabled client consumes 13 seconds averagely over connecting to NFS gateway. In the same way, we observed that Local privileged client consumes 19 seconds averagely when requesting connection to NFS gateway. Similarly, we observe that Local unprivileged client consumes 25 seconds averagely at establishing connection to NFS Gateway. The MEAP-enabled client is 48.19% effective than Local privileged and 94.5% effective than Local unprivileged client over establishing connection to NFS gateway as shown in Figure-4.

4.3.2 Multi-homing network client access

In the same way, we also evaluated the efficiency of Remote network client connectivity again using remote network lookup analysis [21, 22] onto three types of connections i.e. Remote MEAP-NFS enabled client, Remote privileged client, and Remote unprivileged client. MEAP sets HDFS threshold of remote network connections on 500 instances. We observed that Remote MEAP-NFS enabled client consumes 19 seconds averagely for establishing connection with NFS gateway. Similarly, we evaluated that Remote privileged client consumes 28 seconds averagely over establishing connection to NFS gateway. In the same way, we find that Remote unprivileged client consumes 32 seconds averagely while establishing connection with gateway. Remote MEAP-enabled client is 39.43% efficient than Remote privileged and 73.41% effective than Remote

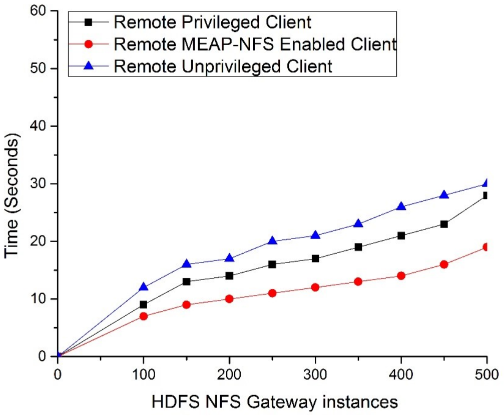


Fig. 5: Multi-homing network clients connectivity over HDFS NFS Gateway

unprivileged client while establishing connection to NFS gateway as shown in Figure-5.

5 Conclusion

This paper proposes a novel Multi-homing key Exchange Authorization Protocol (MEAP) onto the sub-aqueous IoT sensor HDFS repository. The proposed approach introduces a compatible solution to interact multi-homing clients with Hadoop network underwater and finds the clients connectivity reliable by avoiding the available low communication bandwidth. The experimentation is carried out over three connections with respect to local and multi-homing networks and MEAP proven the worth of proposed scheme in sub-aqueous distributed computing environment of Hadoop cluster.

In future, we focus to incorporate other resistance mediums of sub-aqueous environment in multi-homing Hadoop cluster.

References

1. LaValle, Steve, et al. ”Big data, analytics and the path from insights to value.” MIT sloan management review 52.2, 2011, pp. 21.
2. Cloudera, ”The modern platform for data management and analytics,” Cloudera, 2016. [Online]. Available: http://www.cloudera.com/. Accessed: Jun. 05, 2018.
3. M. Technologies, ”Featured customers”, 2016. [Online]. Available: https://www.mapr.com/. Accessed: Jun. 05, 2018.
4. ”Welcome to Apache Hadoop!” 2014. [Online]. Available: http://hadoop.apache.org/. Accessed: Jun. 05, 2018.
5. ”Apache Hadoop 2.7.2 Apache Hadoop YARN,” 2016. [Online]. Available: https://hadoop.apache.org/docs/r2.7.2/hadoop-yarn/hadoop-yarn-site/YARN.html. Accessed: Jun. 05, 2018.

6. "Apache Hadoop 2.7.2 MapReduce Tutorial," 2016. [Online]. Available: https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html. Accessed: Jun. 05, 2018.
7. "Apache Hadoop 2.7.2 HDFS users guide," 2016. [Online]. Available: https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsUserGuide.html. Accessed: Jun. 05, 2018.
8. Neuman, B. Clifford, and Theodore Ts'o. "Kerberos: An authentication service for computer networks." IEEE Communications magazine 32.9 (1994): 33-38.
9. Wang, Guohui, T. S. Ng, and Anees Shaikh. "Programming your network at run-time for big data applications." Proceedings of the first workshop on Hot topics in software defined networks. ACM, 2012.
10. "Apache Hadoop 2.7.2 HDFS NFS gateway," 2016. [Online]. Available: https://hadoop.apache.org/docs/r2.7.2/hadoop-project-dist/hadoop-hdfs/HdfsNfsGateway.html. Accessed: Jun. 05, 2018.
11. QURESHI, FASEEH, et al. "KEY EXCHANGE AUTHENTICATION PROTOCOL FOR NFS ENABLED HDFS CLIENT." Journal of Theoretical & Applied Information Technology 95.7 (2017).
12. A. Alharbi, R. Ammar, H. Alhumyani, S. Rajasekaran and Jun-Hong, "Efficient pipeline architectures for Underwater Big data analytic," 2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Noida, 2014, pp. 000161-000166.
13. J. Cohen, and S. Acharya, "Towards a more secure Apache Hadoop HDFS infrastructure," in Lecture Notes in Computer Science. Springer Nature, 2013, pp. 735-741.
14. A. Boldyreva, "Threshold signatures, Multisignatures and blind signatures based on the Gap-Diffie-Hellman-Group signature scheme," in Public Key Cryptography PKC 2003. Springer Nature, 2002, pp. 31-46.
15. H. Dubner, "Large Sophie Germain primes." Mathematics of Computation of the American Mathematical Society 65.213, 1996, pp. 393-396.
16. D. H. Lehmer, "On Eulers totient function." Bulletin of the American Mathematical Society 38.10, 1932, pp. 745-751.
17. M. Bellare, and S. K. Miner. "A forward-secure digital signature scheme." Annual International Cryptology Conference. Springer Berlin Heidelberg, 1999.
18. N. M. F. Qureshi, and D. R. Shin, "RDP: A storage-tier-aware Robust Data Placement strategy for Hadoop in a Cloud-based Heterogeneous Environment", KSII Transactions on Internet and Information Systems, vol. 10, no. 9, 2016, pp. 4063-4086.
19. "Sub-aqueous IoT Dataset " 2018. [Online]. Available: https://data.world/datasets/sensors. Accessed: Jun. 05, 2018.
20. S. Huang, et al. "The HiBench benchmark suite: Characterization of the MapReduce-based data analysis." New Frontiers in Information and Software as Services. Springer Berlin Heidelberg, 2011, pp. 209-228.
21. S. Mazumder, "Big Data Tools and Platforms." Big Data Concepts, Theories, and Applications. Springer International Publishing, 2016, pp. 29-128.
22. S. Magana-zook, et al. "Large-scale seismic waveform quality metric calculation using Hadoop." Computers & Geosciences, 2016.