IJCSEC- International Journal of Computer Science and Engineering Communications.
Vol.3, Issue 2, 2015, Page.854-861, ISSN: 2347–8586

Remote Data Auditing Scheme in Secured Cloud Storage Environment

Sangeetha.T [1],Saranya.M [2]
PG Scholar[1] , Assistant Professor[2]
Nandha College of Technology, Erode, India
sangee9110@gmail.com [1],saranyamcse88@gmail.com [2]

Abstract: Cloud data centers are used to maintain the shared data values for the data owners .Data owners and public verifiers are involved to efficiently audit cloud data integrity without retrieving the entire data from the cloud server. File and block signatures are used in the integrity verification process. Public data auditing schemes are tuned to verify the encrypted cloud storage environment. "One Ring to RUle Them All" (Oruta) scheme is used for privacy-preserving public auditing process. In oruta homomorphic authenticators are constructed using Ring Signatures. Ring signatures are used to compute verification metadata needed to audit the correctness of shared data. The identity of the signer on each block in shared data is kept private from public verifiers. Homomorphic authenticable ring signature (HARS) scheme is applied to provide identity privacy with blockless verification. Batch auditing mechanism supports to perform multiple auditing tasks simultaneously.Oruta is compatible with random masking to preserve data privacy from public verifiers. Dynamic data management process is handled with index hash tables. Traceability is not supported in oruta scheme. Data dynamism sequence is not managed by the system. The system obtains high computational overhead.Privacy preserved data verification techniques are applied on the cloud data centers to check the encrypted data values. Traceability features are provided with identity privacy. Group manager or data owner can be allowed to reveal the identity of the signer based on verification metadata. Data version management mechanism is integrated with the system.

I INTRODUCTION

Cloud computing has been envisioned as the next generation information technology (IT) architecture for enterprises, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk [2]. As a disruptive technology with profound implications, cloud computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data are being centralized or outsourced to the cloud. From users' perspective, including both individuals and IT enterprises, storing data remotely to the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with location independence and avoidance of capital expenditure on hardware, software and personnel maintenances, etc., [3].

While cloud computing makes these advantages more appealing than ever, it also brings new and challenging security threats toward users' outsourced data. Since cloud service providers (CSP) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. First of all, although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity [4]. Examples of outages and security breaches of noteworthy cloud services appear from time to time. Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status. For examples, CSP might reclaim storage for monetary reasons by discarding data that have not been or are rarely accessed, or even hide data loss incidents to maintain a reputation [8]. In short, although outsourcing data to the cloud is economically attractive for long-term large-scale storage, it does not immediately offer any guarantee on

data integrity and availability. This problem, if not properly addressed, may impede the success of cloud architecture.

As users no longer physically possess the storage of their data, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted. In particular, simply downloading all the data for its integrity verification is not a practical solution due to the expensiveness in I/O and transmission cost across the network. Besides, it is often insufficient to detect the data corruption only when accessing the data, as it does not give users correctness assurance for those unaccessed data and might be too late to recover the data loss or damage Considering the large size of the outsourced data and the user's constrained resource capability, the tasks of auditing the data correctness in a cloud environment can be formidable and expensive for the cloud users [9]. Moreover, the overhead of using cloud storage should be minimized as much as possible, such that a user does not need to perform too many operations to use the data. In particular, users may not want to go through the complexity in verifying the data integrity. Besides, there may be more than one user accesses the same cloud storage, say in an enterprise setting. For easier management, it is desirable that cloud only entertains verification request from a single designated party.

To fully ensure the data integrity and save the cloud users' computation resources as well as online burden, it is of critical importance to enable public auditing service for cloud data storage, so that users may resort to an independent third-party auditor (TPA) to audit the outsourced data when needed. The TPA, who has expertise and capabilities that users do not, can periodically check the integrity of all the data stored in the cloud on behalf of the users, which provides a much more easier and affordable way for the users to ensure their storage correctness in the cloud. Moreover, in addition to help users to evaluate the risk of their subscribed cloud data services, the audit result from TPA would also be beneficial for the cloud service providers to improve their cloud-based service platform and even serve for independent arbitration purposes. In a word, enabling public auditing services will play an important role for this nascent cloud economy to become fully established; where users will need ways to assess risk and gain trust in the cloud.

Recently, the notion of public auditability has been proposed in the context of ensuring remotely stored data integrity under different system and security models. Public auditability allows an external party, in addition to the user himself, to verify the correctness of remotely stored data. Most of these schemes do not consider the privacy protection of users' data against external auditors. Indeed, they may potentially reveal user's data to auditors. This severe drawback greatly affects the security of these protocols in cloud computing. From the perspective of protecting data privacy, the users, who own the data and rely on TPA just for the storage security of their data, do not want this auditing process introducing new vulnerabilities of unauthorized information leakage toward their data security [10]. Moreover, there are legal regulations, such as the US Health Insurance Portability and Accountability Act (HIPAA), further demanding the outsourced data not to be leaked to external parties. Simply exploiting data encryption before outsourcing could be one way to mitigate this privacy concern of data auditing, but it could also be overkill when employed in the case of unencrypted/public cloud data, due to the unnecessary processing burden for cloud users. Besides, encryption does not completely solve the problem of protecting data privacy against third-party auditing but just reduces it to the complex key management domain. Unauthorized data leakage still remains possible due to the potential exposure of decryption keys.

Therefore, how to enable a privacy-preserving third party auditing protocol, independent to data encryption, is the problem we are going to tackle in this paper. Our work is among the first few ones to support privacy-preserving public auditing in cloud computing, with a focus on data storage. Besides, with the prevalence of cloud computing, a foreseeable increase of auditing tasks from different users may be delegated to TPA.

II RELATED WORK

Recently, much of growing interest has been pursued in the context of remotely stored data verification [5]. Ateniese et al. are the first to consider public auditability in their defined "provable data possession" model for ensuring possession of files on untrusted storages. In their scheme, they utilize RSA-based homomorphic tags for auditing outsourced data, thus public auditability is achieved. Ateniese et al. do not consider the case of dynamic data storage and the direct extension of their scheme from static data storage to dynamic case may suffer design and security problems. In their subsequent work, Ateniese et al. propose a dynamic version of the prior PDP scheme. The system imposes a priori bound on the number of queries and does not support fully dynamic data operations, i.e., it only allows very basic block operations with limited functionality and block insertions cannot be supported. In [6], Wang et al. consider dynamic data storage in a distributed scenario and the proposed challenge-response protocol can both determine the data correctness and locate possible errors.

They only consider partial support for dynamic data operation. Juels and Kaliski describe a "proof of retrievability" model, where spot-checking and error-correcting codes are used to ensure both "possession" and "retrievability" of data files on archive service systems. Specifically, some special blocks called "sentinels" are randomly embedded into the data file F for detection purpose and F is further encrypted to protect the positions of these special blocks. The number of queries a client can perform is also a fixed priori and the introduction of precomputed "sentinels" prevents the development of realizing dynamic data updates. In addition, public auditability is not supported in their scheme. Shacham and Waters design an improved PoR scheme with full proofs of security in the security model. They use publicly verifiable homomorphic authenticators built from BLS signatures based on which the proofs can be aggregated into a small authenticator value and public retrievability is achieved. Still, the authors only consider static data files. Erway et al. [7] were the first to explore constructions for dynamic provable data possession. They extend the PDP model support provable updates to stored data files using rank-based authenticated skip lists. This scheme is essentially a fully dynamic version of the PDP solution. To support updates, especially for block insertion, they eliminate the index information in the "tag" computation in Ateniese's PDP model and employ authenticated skip list data structure to authenticate the tag information of challenged or updated blocks first before the verification procedure. The efficiency of their scheme remains unclear.

Although the existing schemes aim at providing integrity verification for different data storage systems, the problem of supporting both public auditability and data dynamics has not been fully addressed. How to achieve a secure and efficient design to seamlessly integrate these two important components for data storage service remains an open challenging task in Cloud Computing.

Portions of the work presented in this paper have previously appeared as an extended abstract. We revise the paper a lot and add more technical details. First, before the introduction of our proposed construction, we present two basic solutions for realizing data auditability and discuss their demerits in supporting public auditability and data dynamics. Second, we generalize the support of data dynamics to both PoR and PDP models and discuss the impact of dynamic data operations on the overall system efficiency both. In particular, we emphasize that while dynamic data updates can be performed efficiently in PDP models more efficient protocols need to be designed for the update of the encoded files in PoR models. We extend our data auditing scheme for the single client and explicitly include a concrete description of the multi client data auditing scheme. We also redo the whole experiments and present the performance comparison between the multi client data auditing scheme and the individual auditing scheme. Finally, for the proposed theorems in this paper, we provide formal security proofs under the random oracle model, which are lacking in [1].

III ACCOUNTABILITY AND PRIVACY PROTECTION

Accountability in our sense will be achieved via a combination of private and public accountability. Public accountability is derived from an active interaction between: subjects of PII; regulatory bodies, such as Information Commissioners; data controllers. It is premised upon highly transparent processes. Private accountability, in contrast, is derived from the interaction between data controllers and data processors and is premised on contract law, technological processes and practical internal compliance requirements. The objective of such accountability is not to meet 'a set of largely procedural requirements for … processing activities' but rather to reduce the risk of disproportionate harm to the subjects of PII and thus reduce or permit the amelioration of negative consequences for the data controller. It reflects an acceptance that absolute reduction of harm to the subjects of PII is an impossible goal in a disaggregated environment, such as a cloud service and that the ability to respond flexibly and efficiently to harms arising will provide a more efficient form of privacy protection than enforcing blunt and/or static 'tick-box' compliance criteria.

Solutions to privacy risks in the cloud involve reintroducing an element of control. For the corporate user, privacy risk in cloud computing can be reduced if organizations involved in cloud provision use a combination of privacy policies and contractual terms to create accountability in the form of transparent, enforceable commitments to responsible data handling. Specifically, accountable organizations will ensure that obligations to protect data are observed by all processors of the data, irrespective of where that processing occurs.

Through contractual agreements, all organizations involved in the cloud provision would be accountable. While the corporate user, as the first corporate entity in the cloud provision, would be held legally accountable, the corporate user would then hold the initial service provider (SP1) accountable through contractual agreements, requiring in turn that SP1 hold its SPs accountable contractually as well. This is analogous to some existing cases in outsourcing environments, where the transferor is held accountable by regulators even when it is the transferee that does not act in accordance with individuals' wishes. The following elements are key to provision of accountability within the cloud:

3.1. Transparency

Individuals should be adequately informed about how their data is handled within the cloud and the responsibilities of people and organizations in relation to the processing of PII should be clearly identified. As with other disaggregated data environments, transparency in cloud computing is important not only for legal and regulatory reasons, but also to avoid violation of social norms. In the context of this paper, transparency means a level of openness about an entity's handling of PII that permits meaningful accountability.

3.2. Assurance

The corporate user provides assurance and transparency to the customer/client through its privacy policy, while requiring similar assurances from the SP through contractual measures and audits.

3.3. User Trust

Accountability helps foster user trust. When it is not clear to individuals why their personal information is requested, or how and by whom it will be processed, this lack of control will lead to suspicion and ultimately distrust. There are also security-related concerns about whether data in the cloud will be adequately protected.

3.4. Responsibility

Most data protection regimes require a clear allocation of responsibility for the processing of PII, as existing regulatory mechanisms rely heavily upon user and regulator intervention with responsible

parties. Disaggregated data environments, e.g. mobile e-commerce and cloud computing, can hinder determination of that responsibility. Predetermining responsibility, via contract, as information is shared and processed within the cloud, pre-empts perceptions of regulatory failure, which may erode user trust. It also permits companies to assess their trading risks in terms of potential financial losses and data privacy breaches.

3.5. Policy Compliance

Accountability helps ensure that the cloud service complies with laws and also the mechanisms proposed in this paper help compliance with cloud provider organizational policies and user preferences and with auditing.

IV REMOTE DATA AUDITING SCHEME FOR CLOUDS

Cloud service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches. It is routine for users to leverage cloud storage services to share data with others in a group, as data sharing becomes a standard feature in most cloud storage offerings, including Dropbox, iCloud and Google Drive. The integrity of data in cloud storage, however, is subject to skepticism and scrutiny, as data stored in the cloud can easily be lost or corrupted due to the inevitable hardware/ software failures and human errors. To make this matter even worse, cloud service providers may be reluctant to inform users about these data errors in order to maintain the reputation of their services and avoid losing profits. The integrity of cloud data should be verified before any data utilization, such as search or computation over cloud data.

The traditional approach for checking data correctness is to retrieve the entire data from the cloud and then verify data integrity by checking the correctness of signatures of the entire data. Certainly, this conventional approach is able to successfully check the correctness of cloud data. The efficiency of using this traditional approach on cloud data is in doubt. We believe that sharing data among multiple users is perhaps one of the most engaging features that motivates cloud storage. Therefore, it is also necessary to ensure the integrity of shared data in the cloud is correct. Existing public auditing mechanisms can actually be extended to verify shared data integrity. A new significant privacy issue introduced in the case of shared data with the use of existing mechanisms is the leakage of identity privacy to public verifiers.

For instance, Alice and Bob work together as a group and share a file in the cloud. The shared file is divided into a number of small blocks, where each block is independently signed by one of the two users with existing public auditing solutions. Once a block in this shared file is modified by a user, this user needs to sign the new block using his/her private key [11]. Eventually, different blocks are signed by different users due to the modification introduced by these two different users. Then, in order to correctly audit the integrity of the entire data, a public verifier needs to choose the appropriate public key for each block. As a result, this public verifier will inevitably learn the identity of the signer on each block due to the unique binding between an identity and a public key via digital certificates under public key infrastructure (PKI).

Failing to preserve identity privacy on shared data during public auditing will reveal significant confidential information to public verifiers. Specifically, after performing several auditing tasks, this public verifier can first learn that Alice may be a more important role in the group because most of the blocks in the shared file are always signed by Alice; on the other hand, this public verifier can also easily deduce that the eighth block may contain data of a higher value, because this block is frequently modified by the two different users. In order to protect these confidential information, it is essential and critical to preserve identity privacy from public verifiers during public auditing.

In this paper, to solve the above privacy issue on shared data, we propose Oruta, a novel privacy-preserving public auditing mechanism. More specifically, we utilize ring signatures to construct homomorphic authenticators in Oruta, so that a public verifier is able to verify the integrity of shared data

without retrieving the entire data—while the identity of the signer on each block in shared data is kept private from the public verifier. In addition, we further extend our mechanism to support batch auditing, which can perform multiple auditing tasks simultaneously and improve the efficiency of verification for multiple auditing tasks. Meanwhile, Oruta is compatible with random masking, which has been utilized in WWRL and can preserve data privacy from public verifiers. Moreover, we also leverage index hash tables from a previous public auditing solution to support dynamic data. A high-level comparison among Oruta and existing mechanisms is presented.

V PROBLEM FORMULATION

"One Ring to RUle Them All" (Oruta) scheme is used for privacy-preserving public auditing process. In oruta homomorphic authenticators are constructed using Ring Signatures. Ring signatures are used to compute verification metadata needed to audit the correctness of shared data. The identity of the signer on each block in shared data is kept private from public verifiers. Homomorphic authenticable ring signature (HARS) scheme is applied to provide identity privacy with blockless verification. Batch auditing mechanism supports to perform multiple auditing tasks simultaneously. Oruta is compatible with random masking to preserve data privacy from public verifiers. Dynamic data management process is handled with index hash tables. The following problems are identified from the existing system. They are traceability is not supported, data dynamism sequence is not managed and high computational overhead.

VI ORUTA SCHEME FOR PRIVACY PRESERVED CLOUD DATA ANALYSIS

The system model in this paper involves three parties: the cloud server, a group of users and a public verifier. There are two types of users in a group: the original user and a number of group users. The original user initially creates shared data in the cloud and shares it with group users. Both the original user and group users are members of the group. Every member of the group is allowed to access and modify shared data. Shared data and its verification metadata are both stored in the cloud server. A public verifier, such as a third-party auditor providing expert data auditing services or a data user outside the group intending to utilize shared data, is able to publicly verify the integrity of shared data stored in the cloud server. When a public verifier wishes to check the integrity of shared data, it first sends an auditing challenge to the cloud server. After receiving the auditing challenge, the cloud server responds to the public verifier with an auditing proof of the possession of shared data. Then, this public verifier checks the correctness of the entire data by verifying the correctness of the auditing proof.

Two kinds of threats related to the integrity of shared data are possible. First, an adversary may try to corrupt the integrity of shared data. Second, the cloud service provider may inadvertently corrupt data in its storage due to hardware failures and human errors. Making matters worse, the cloud service provider is economically motivated, which means it may be reluctant to inform users about such corruption of data in order to save its reputation and avoid losing profits of its services. The identity of the signer on each block in shared data is private and confidential to the group. Once the public verifier reveals the identity of the signer on each block, it can easily distinguish a high-value target from others.

Oruta should be designed to achieve following properties: (1) Public Auditing: A public verifier is able to publicly verify the integrity of shared data without retrieving the entire data from the cloud. (2) Correctness: A public verifier is able to correctly verify shared data integrity. (3) Unforgeability: Only a user in the group can generate valid verification metadata on shared data. (4) Identity Privacy: A public verifier cannot distinguish the identity of the signer on each block in shared data during the process of auditing.

We intend to utilize ring signatures to hide the identity of the signer on each block, so that private and sensitive information of the group is not disclosed to public verifiers. Traditional ring signatures cannot be directly used into public auditing mechanisms.Without blockless verifiability, a public verifier has to download the whole data file to verify the correctness of shared data, which consumes excessive

bandwidth and takes very long verification times. We design a new homomorphic authenticable ring signature (HARS) scheme, which is extended from a classic ring signature scheme. The ring signatures generated by HARS are not only able to preserve identity privacy but also able to support blockless verifiability.

VII REMOTE DATA AUDITING SCHEME IN SECURED CLOUDS

The proposed system is designed to perform public data verification with privacy. Traceability features are provided with identity privacy. Group manager or data owner can be allowed to reveal the identity of the signer based on verification metadata. Data version management mechanism is integrated with the system. The system is divided in to five major modules. They are data center, third party auditor, client data dynamism handler and batch auditing. The cloud data center manages the shared data values. Auditing operations are initiated by the Third Party Auditor. Client application is designed to manage data upload and download operations. Data update operations are managed under data dynamism module. Batch auditing is designed for multi user data verification process.

7.1. Data Center

The data center application is designed to allocate storage space for the data providers. Data center maintains data files for multiple providers. Different sized storage area is allocated for the data providers. Data files are delivered to the clients.

7.2. Third Party Auditor

The Third Party Auditor (TPA) maintains the signature for shared data files. TPA performs the public data verification for data providers. Data integrity verification is performed is using Secure Hashing Algorithm (SHA). Homomorphic linear authenticator and random masking techniques are used for privacy preservation process.

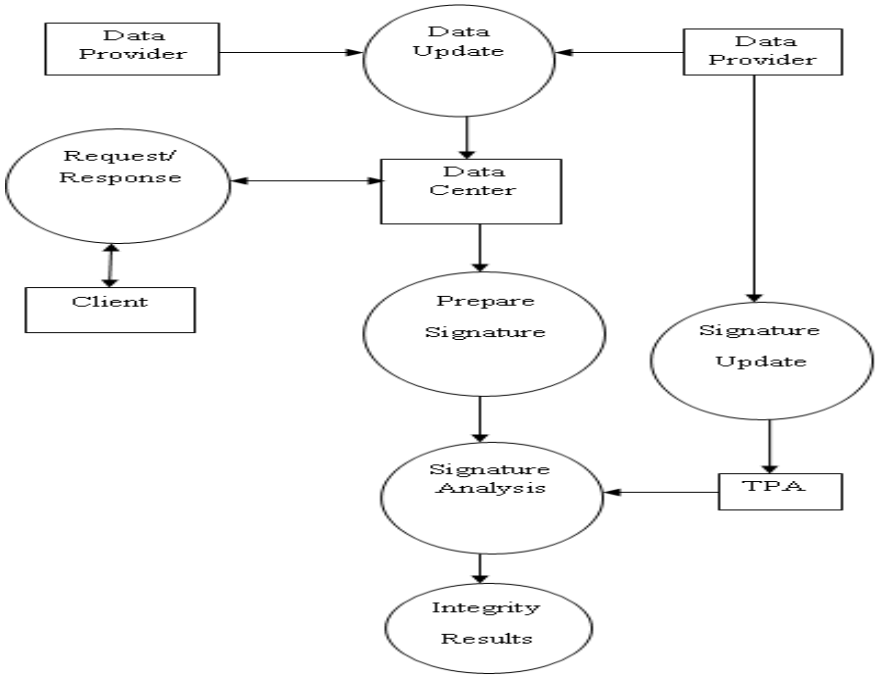

Fig. No: 7.1. Remote Data Auditing Scheme in Secured Clouds

7.3. Client

The client application is designed to access the hard data values. The cloud user initiates the download process. Data access information is updated to the data center. Data center transfers the data as blocks.

7.4. Data Dynamism Handler

Shared data values are managed with blocks. Block update and delete operations are handled with signature update process. Block insertion operations are also supported in data dynamism process. Block signatures are also updated in data dynamism process.

7.5. Batch Auditing

Data integrity verification is carried out under auditing process. Batch auditing is applied to perform simultaneous data verification process. Batch auditing is tuned for multi user environment. Data dynamism is integrated with batch auditing process.

VIII CONCLUSION

Public auditing schemes are used to verify the data integrity in cloud servers. Oruta (One Ring to Rule Them All) scheme is used to support privacy ensured data verification process. Data dynamism and batch auditing are supported in Oruta. Oruta scheme is enhanced with Traceability and Data freshness features. Privacy ensured data verification is performed. Simultaneous data verification scheme is provided in the system. Computational and communication cost is reduced by the system. The system supports data dynamism for secured cloud storage environment. Traceability and version management mechanism is integrated with the system.

REFERENCES

[1] Q. Wang, C. Wang, J. Li, K. Ren and W. Lou, "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing," Proc. 14th European Symp. Research in Computer Security, pp. 355-370, 2009.
[2] P. Mell and T. Grance, "Draft NIST Working Definition of Cloud Computing," http://csrc.nist.gov/groups/SNS/cloudcomputing/index.html, June 2009.
[3] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, A. Rabkin, I. Stoica and M. Zaharia, "Above the Clouds: A Berkeley View of Cloud Computing," Technical Report UCB-EECS-2009-28, Univ. of California, Berkeley, Feb. 2009.
[4] Cloud Security Alliance, "Top Threats to Cloud Computing," http://www.cloudsecurityalliance.org, 2010.
[5] K.D. Bowers and A. Oprea, "Hail: A High-Availability and Integrity Layer for Cloud Storage," Proc. 16th ACM Conf. Computer and Comm. Security, 2009.
[6] C. Wang, Q. Wang, K. Ren and W. Lou, "Ensuring Data Storage Security in Cloud Computing," Proc. 17th Int'l Workshop Quality of Service, 2009.
[7] C. Erway, A. Kupcu, C. Papamanthou and R. Tamassia, "Dynamic Provable Data Possession," Proc. 16th ACM Conf. Computer and Comm. Security, 2009.
[8] Q. Wang, C. Wang, K. Ren and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.
[9] Cloud Security Alliance, "Security Guidance for Critical Areas of Focus in Cloud Computing," http://www.cloudsecurityalliance. org, 2009.
[10] C. Wang, K. Ren, W. Lou and J. Li, "Towards Publicly Auditable Secure Cloud Data Storage Services," IEEE Network Magazine, July/Aug. 2010.
[11] Yuan Yao and Michael J. Neely, "Power Cost Reduction in Distributed Data Centers: A Two-Time-Scale Approach for Delay Tolerant Workloads", IEEE Transactions On Parallel And Distributed Systems, January 2014.