Volume 5, Issue 2, February 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Hybrid Approach for Cloud Bandwidth and Cost Reduction System Using Predictive Acknowledgements

Syed Sajid Hussain
M.Tech CSE, Bharath University
Chennai, India

A R Arunachalam
Associate Professor, Bharath University,
Chennai, India

Abstract--- ***During the last few years there has been a tremendous and efficient improvement in the computational world and one of the area that is leaving its footprints is cloud computing. One of the main trending issue in this computing is traffic redundancy elimination (TRE). Cloud-based TRE needs to apply a judicious use of cloud resources so that the bandwidth cost reduction is optimized together with the additional cost of TRE computation. Here, I present an approach that is suitable for cloud computing customers called Hybrid PACK(predictive acknowledgements) regarding TRE systems. Unlike other techniques PACK's main advantage is its capability of offloading the cloud-server TRE effort to the end clients , and hence minimizing the overhead and processing costs induced by the TRE algorithm. In the present scenario the server keeps track of all the end clients. But this is not the case with the PACK as in this mechanism a client can manage his own status and hence the server is offloaded. This makes the PACK suitable for pervasive computing. I here present a hybrid implementation of the PACK in which there is an active involvement of client and server and moreover the chunking signatures are done using SHA 2 (secure hash algorithm) in order to process the any size of chunks.***

Keywords--- Predictive Acknowledgement, Traffic Redundancy Elimination, Caching, Cloud Computing, Network Optimization.

I. INTRODUCTION

Cloud computing is emerging style of delivery in which applications, data and resources are rapidly provisioned as standardized offerings to users with a flexible price. The cloud computing paradigm has achieved widespread adoption in recent years. Its success is due largely to customer's ability to use services on demand with a pay-as-you go pricing [5] model, which has proved convenient in many respects. Low costs and high flexibility make migrating to the cloud compelling. Cloud computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. Traffic redundancy and elimination approach is used for minimizing [5] the cost. Traffic redundancy stems from common end-users' activities, such as repeatedly accessing, downloading, distributing and modifying the same or similar information items (documents, data, web and video). TRE is used to eliminate the transmission of redundant content and, therefore, to significantly reduce the network cost. In most common TRE solutions, both the sender and the receiver ex- amine and compare signatures of data chunks, parsed according to the data content prior to their transmission. When redundant chunks are detected, the sender replaces the transmission of each redundant chunk with its strong signature [2]. Commercial TRE solutions are popular at enterprise networks, and involve the deployment of two or more proprietary protocol, state synchronized middle-boxes at both the intranet entry points of data centres and branch offices, eliminating repetitive traffic between them [4], [9] (e.g., Cisco , Riverbed , Quantum , Juniper , Blue- coat , Expand Networks and F5). While proprietary middle-boxes are popular point solutions within enterprises, they are not as attractive in a cloud environment. First, cloud providers cannot benefit from a technology whose goal is to reduce customer bandwidth bills, and thus are not likely to invest in one. Moreover, a fixed client-side and server-side middle-box pair solution is inefficient for a combination of a mobile environment, which detaches the client from a fixed location, and cloud-side elasticity which motivates work distribution and migration among data centres. Therefore, it is commonly agreed that a universal , software-based , end-to-end TRE is crucial in today's pervasive environment. This enables the use of a standard protocol stack and makes a TRE within end-to-end secured traffic (e.g., SSL) possible.

In this paper we proposed a new method called hybrid approach for cloud bandwidth and cost reduction system using predictive acknowledgements.

II. RELATED WORK

There has been an enormous and steady usage of several TRE techniques. A protocol independent technique for eliminating redundant network traffic was proposed in [1]. The paper describes a packet-level TRE, utilizing the algorithms presented [5]. The concept of a low-bandwidth network file system [2] was implemented. Users rarely consider running network file systems over slow or wide-area networks, as the performance would be unacceptable and the bandwidth consumption is too high. A network file system designed for low-bandwidth networks (LBFS) exploits similarities between files or versions of the same file to save bandwidth. It avoids sending data over the network when the same data can already be found in the server's file system or the client's cache. Using this technique in conjunction with conventional compression and caching, LBFS consumes over an order of magnitude less bandwidth than traditional network file systems on common workloads. With the advent of the cloud computing [5]. Developers with innovative ideas for new internet services no longer require the large capital outlays in hardware to deploy their service or the human expense to operate it. They need not be concerned about over provisioning for a service whose popularity does not meet their predictions, thus wasting costly resources, or under provisioning for one that becomes wildly popular, thus missing potential customers and revenue.

In last couple of years there huge increase in the usage cloud computing because cloud computing is emerging style of IT-delivery in which applications, data and resources are rapidly provisioned provided as Standardized offerings to users with a flexible price. But it is important to provide the convenient pricing model for the users of cloud. Hence, a new traffic redundancy and elimination scheme [6] has been designed for reducing the cloud bandwidth and costs. This new traffic redundancy elimination approach also called as novel-TRE or receiver based TRE, which detects redundancy at the client side. However for server specific TRE approach it is difficult to handle the traffic efficiently and it doesn't suites for the cloud environment because of high processing costs.

Cloud computing provides users to store their data remotely and enjoy the on demand high quality services and offers it as usage based pricing. Here [7] , Cloud service providers (CSP) are trying to reduce this bandwidth and also applying the judicious use of cloud resources by deploying a TRE system to end-to-end clients. Despite the exceptional prominence [8] of the cloud computing , the customers are lack of direct sense to select the cloud that delivers the best performance, due to the Performance heterogeneity of each cloud provider. Existing solutions either migrate the application to each cloud and evaluate the performance individually, or benchmark each cloud along various dimensions and predict the overall performance of the application.

A large scale study of real-life traffic redundancy is presented in [12], [13], [14]. Our paper aims at on their finding that an end-to-end redundancy elimination solution, could obtain most of middle-box's bandwidth savings, motivating the benefit of low cost software end-to-end solutions.

Wanax [15] is a TRE system for the developing world where storage and WAN bandwidth are scarce. It is a software-based middle-box replacement for the expensive commercial hardware. In this scheme, the sender middle-box holds back the TCP stream and sends data signatures to receiver middle-box. The receiver checks whether the data is found in its local cache. Data chunks that are not found in the cache are fetched from the sender middle-box or a nearby receiver middle-box. Naturally, such a scheme incurs a three-way-handshake latency for noncached data.

EndRE [16] is a sender based end to end TRE for enterprise networks. It uses a new chunking scheme that is faster than the commonly used Rabin fingerprint, but is restricted to chunks as small as 32-64 B. Unlike PACK, EndRE requires the server to maintain a fully and reliably synchronized cache for each client.

III. PROBLEM IDENTIFICATION

The objective of this section is two fold: evaluating the potential data redundancy for several applications that are likely to reside in a cloud, and to estimate the Hybrid PACK performance and cloud costs of the redundancy elimination process.

This project aims on the finding that "an end to end redundancy elimination solution, could obtain most of the middle-box's bandwidth savings," motivating the benefit of low cost software end-to-end solutions. To the best of our knowledge, none of the previous works have addressed the requirements for a cloud-computing-friendly, end-to-end TRE, which forms Hybrid PACK's focus.

IV. EXISTING SYSTEM

Traffic redundancy stems from common end-users' activities, such as repeatedly accessing, downloading, uploading (i.e., backup), distributing, and modifying the same or similar information items (documents, data, Web, and video). TRE is used to eliminate the transmission of redundant content and, therefore, to significantly reduce the network cost. In most common TRE solutions, both the sender and the receiver examine and compare signatures of data chunks, parsed according to the data content, prior to their transmission. When redundant chunks are detected, the sender replaces the transmission of each redundant chunk with its strong signature. Commercial TRE solutions are popular at enterprise networks, and involve the deployment of two or more proprietary-protocol, state synchronized middle-boxes at both the intranet entry points of data centers.

DISADVANTAGES OF EXISTING SYSTEM

- The rise of "on-demand" work spaces, meeting rooms, and work-from-home solutions detaches the workers from their offices. In such a dynamic work environment, fixed-point solutions that require a client-side and a server-side middle-box pair become ineffective.
- Cloud load balancing and power optimizations may lead to a server-side process and data migration environment, in which TRE solutions that require full synchronization between the server and the client are hard to accomplish or may lose efficiency due to lost synchronization
- Current end-to-end solutions also suffer from the requirement to maintain end-to-end synchronization that may result in degraded TRE efficiency.

V. PROPOSED SYSTEM

The proposed system that is an enhanced version of PACK .In this system there is an active participation of server as well as client in a cloud environment. In this solution, each receiver observes the incoming stream and tries to match its chunks with a previously received chunk chain or a chunk chain of a local file. Using the long-term chunk's metadata information kept locally, the receiver sends to the server predictions that include chunks' signatures and easy-to-verify hints of the sender's future data. On the receiver side, a new computationally lightweight chunking (fingerprinting) scheme termed PACK chunking is proposed. PACK chunking is a new alternative for Rabin fingerprinting traditionally used by RE applications.

On the sender's side there is also the provision of PACK chunking inserted in the transport layer (TCP/SSL).whenever the receiver or the client is not able to maintain its own status the server will take care of it by generating a trigger.

VI. ARCHITECTURAL DESIGN SPECIFICATION

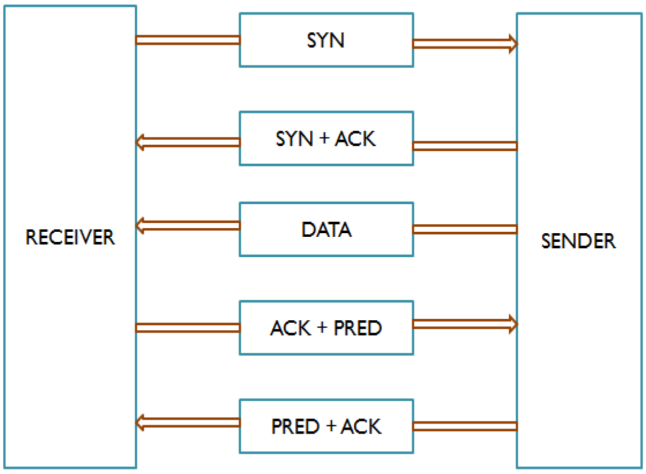


Figure 1. Overall Architecture Of Hybrid PACK

The figure 1 above shows the architecture of a Hybrid PACK. In order to conform to existing firewalls and minimize overheads, we use the TCP options field to carry the PACK wire protocol. It is clear that novel-TRE can also be implemented above the TCP level while using similar message types and control fields. The above Figure illustrates way the novel-TRE operates under the assumption that the data is redundant. First, both sides enable the PACK option during the initial TCP handshake by adding a *PACK* permitted flag to the TCP options field. Then, the sender sends the (redundant) data in one or more TCP segments, and the receiver identifies that a currently received chunk is identical to a chunk in its chunk store. The receiver, in turn, triggers a TCP ACK message and includes the prediction in the packet's options field. Last, the sender sends a confirmation message PRED-ACK replacing the actual data.

VII. MODULE DESCRIPTION

There are three modules in this system:

- Receiver Chunk Store
- Hybrid Algorithm
- Wire Protocol

A. Receiver Chunk Store

PACK uses a new *chains* scheme, which chunks are linked to other chunks according to their last received order. The PACK receiver maintains a *chunk store*, which is a large size cache of chunks and their associated metadata. Chunk's metadata includes the chunk's signature and a (single) pointer to the successive chunk in the last received stream containing this chunk. Caching and indexing techniques are employed to efficiently maintain and retrieve the stored chunks, their signatures, and the chains formed by traversing the chunk pointers.

When the new data are received and parsed to chunks, the receiver computes each chunk's signature using SHA-2. At this point, the chunk and its signature are added to the chunk store. In addition, the metadata of the previously received chunk in the same stream is updated to point to the current chunk. The unsynchronized nature of PACK allows the receiver to map each existing file in the local file system to a chain of chunks, saving in the chunk store only the metadata associated with the chunks.

B. Hybrid Algorithm

Upon the arrival of new data, the receiver computes the respective signature for each chunk and looks for a match in its local chunk store. If the chunk's signature is found, the receiver determines whether it is a part of a formerly received chain, using the chunks' metadata. If affirmative, the receiver sends a prediction to the sender for several next expected chain chunks. Upon a successful prediction, the sender responds with a PRED-ACK confirmation message. Once the PRED-ACK message is received and processed, the receiver copies the corresponding data from the chunk store to its TCP input buffers, placing it according to the corresponding sequence numbers. At this point, the receiver sends a normal TCP ACK with the next expected TCP sequence number. In case the prediction is false, or one or more predicted chunks are already sent, the sender continues with normal operation, e.g., sending the raw data, without sending a PRED-ACK message.

When a sender receives a PRED message from the receiver, it tries to match the received predictions to its buffered (yet to be sent) data. For each prediction, the sender determines the corresponding TCP sequence range and verifies the hint. Upon a hint match, the sender calculates the more computationally intensive SHA-2 signature for the predicted data range and compares the result to the signature received in the PRED message. Note that in case the hint does not match, a computationally expansive operation is saved. If the two SHA-2 signatures match, the sender can safely assume that the receiver's prediction is correct. In this case, it replaces the corresponding outgoing buffered data with a PRED-ACK message.

Fig. 2 illustrates the sender operation using state machines. Fig. 2(a) describes the parsing of received PRED command. Fig. 2(b) describes how the sender attempts to match a predicted range to its outgoing data.

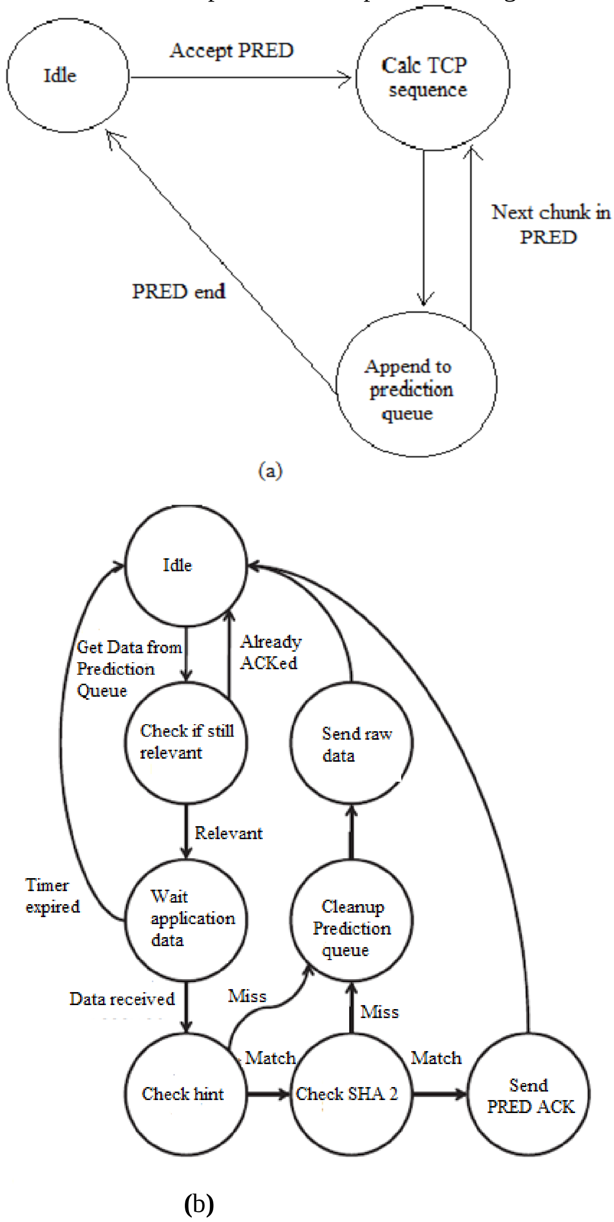


Figure 2. Sender algorithms. (a) Filling the prediction queue ,(b) Processing the prediction queue

C. Wire Protocol

In order to conform with existing firewalls and minimize overheads; we use the TCP Options field to carry the PACK wire protocol. It is clear that PACK can also be implemented above the TCP level while using similar message types and control fields. The PACK wire protocol operates under the assumption that the data is redundant. First, both sides enable the PACK option during the initial TCP handshake by adding a *PACK permitted* to the TCP Options field. Then, the sender sends the (redundant) data in one or more TCP segments, and the receiver identifies that a currently received chunk is identical to a chunk in its chunk store. The receiver, in turn, triggers a TCP ACK message and includes the prediction in the packet's Options field. Last, the sender sends a confirmation message (PRED-ACK) replacing the actual data.

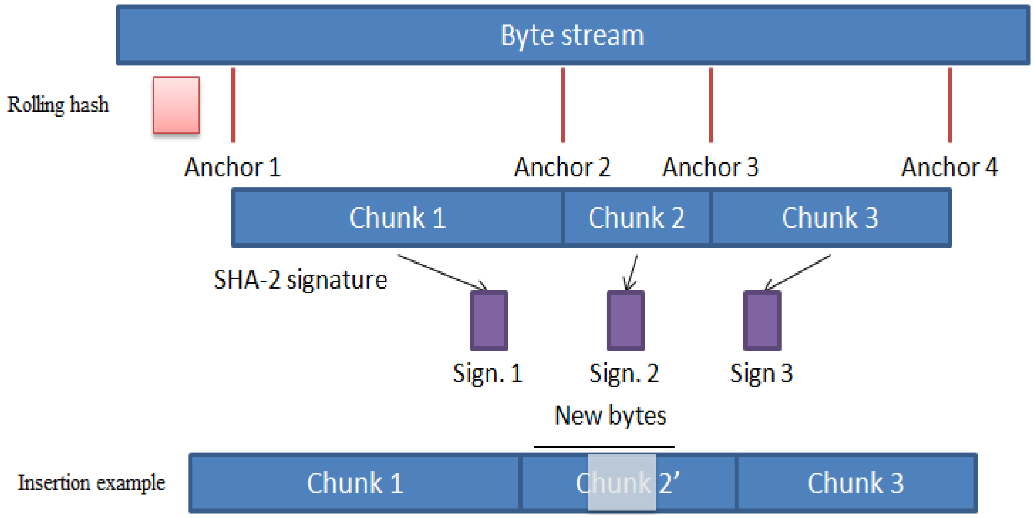
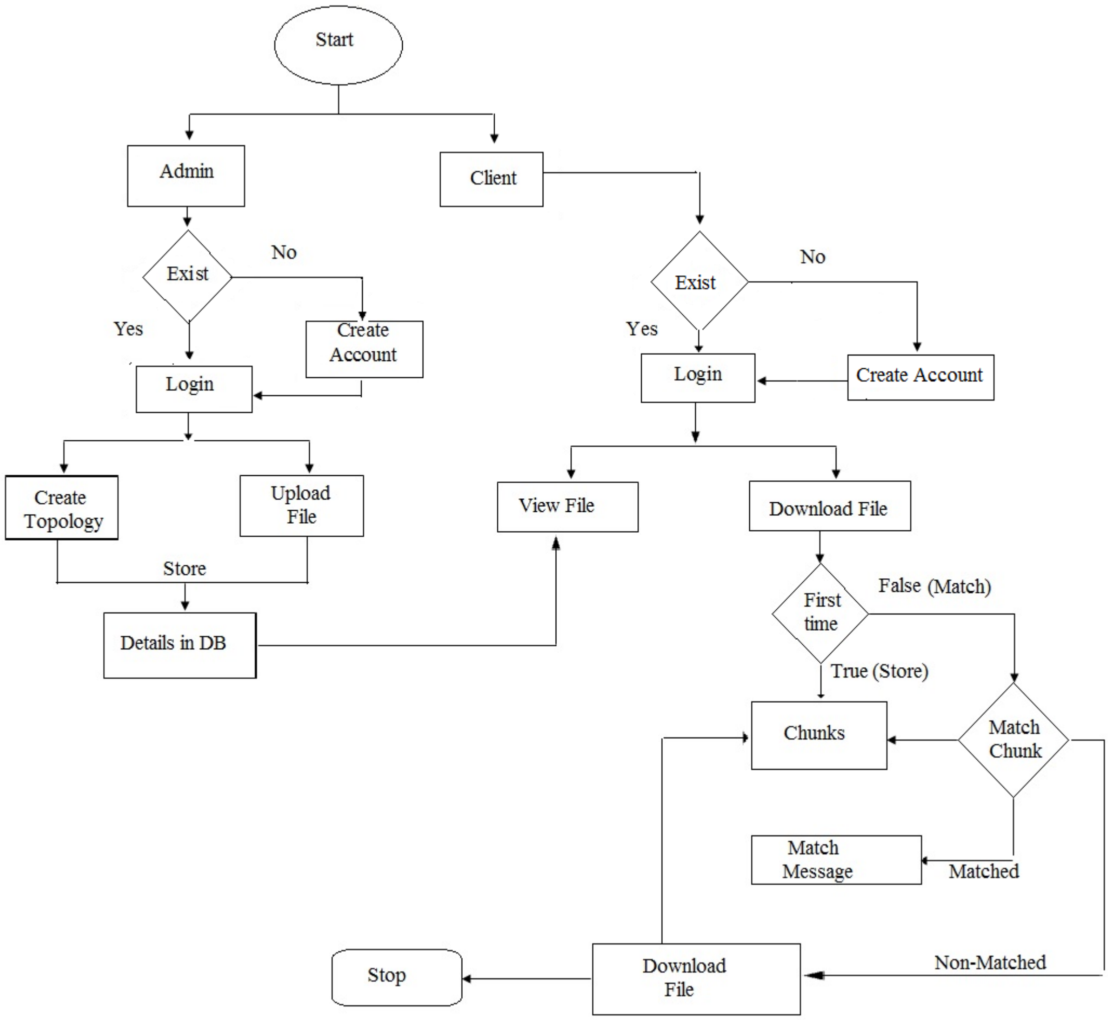


Figure 3. Server parses the outgoing stream to content-based chunks and signs with SHA-2.

VIII. FLOW CHART



IX. CONCLUSION

Cloud computing is expected to trigger high demand for TRE solutions as the amount of data exchanged between the cloud and its users is expected to dramatically increase. The cloud environment redefines the TRE system requirements, making proprietary middlebox solutions inadequate. Consequently, there is a rising need for a TRE solution that reduces the cloud's operational cost while accounting for application latencies, user mobility and cloud elasticity.

Here , PACK, a hybrid-based, cloud-friendly, end-to-end TRE that is based on novel speculative principles that reduce latency and cloud operational cost. PACK does not require the server to continuously maintain clients' status, thus enabling cloud elasticity and user mobility while preserving long-term redundancy. Moreover, PACK is capable of eliminating redundancy based on content arriving to the client from multiple servers without applying a three-way handshake.

The evaluation using a wide collection of content types shows that PACK meets the expected design goals and has clear advantages over sender-based TRE, as well as receiver based especially when the cloud computation cost and buffering requirements are important. Moreover, PACK imposes additional effort on the sender only when redundancy is exploited, thus reducing the cloud overall cost.

Two interesting future extensions can provide additional benefits to the PACK concept. First, the implementation maintains chains by keeping for any chunk only the last observed subsequent chunk in an LRU fashion. An interesting extension to this work is the statistical study of chains of chunks that would enable multiple possibilities in both the chunk order and the corresponding predictions. The system may also allow making more than one prediction at a time, and it is enough that one of them will be correct for successful traffic elimination.

REFERENCES

[1] N. T. Spring and D. Wetherall, "A protocol-independent technique for eliminating redundant network traffic," in Proc. SIGCOMM, 2000, vol.30, pp. 87–95.

[2] A.Muthitacharoen, B. Chen, and D. Mazières, "A low-bandwidth network file system," inProc. SOSP, 2001, pp. 174–187.

[3] Katrina LaCurts, Jeffrey C. Mogul, Hari Balakrishnan, and Yoshio Turner, "Cicada: Predictive Gaurantees for cloud network bandwidth," MIT-CSAIL-TR-2014-004,March 24,2014

[4] E.Lev-Ran, I.Cidon andI.Z.Ben-Shaul, "Methodandapparatus for reducing network traffic over low bandwidth links," US Patent 7636767, Nov. 2009.

[5] M. Armbrust, A. Fox, R. Griffith, A.D.Joseph, R.Katz, A.Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, 2010.

[6] Suresh Chougala and Sharavana K," Design of traffic redundancy and elimination approach for reducing cloud bandwidth and cost," IJIRCCE, Vol. 2,issue 2, February 2014.

[7] Prakash E J, A Nageshwara Rao," Periodically predicting client's bandwidth and cost acknowledgements sends to server to optimize resource usage," IJIRCCE, Vol.2, special issue 4,September 2014.

[8] Xaunran Zong, "Predicting application performance in the cloud," Master's thesis , Duke university,2011.

[9] Rushil Dave, "Mobile virtual network operator systems on cloud: An architectural and cost-benefit study," Master's thesis, august 8,2011.

[10] S.Mccanne and M.Demmer, "Content-based segmentation scheme for data compression in storage and transmission including hierarchical segment representation," US Patent 6828925, Dec. 2004.

[11] E. Zohar, I. Cidon, and O. Mokryn, "The power of prediction: Cloud bandwidth and cost reduction," in *Proc. SIGCOMM*, 2011, pp. 86–97.

[12] A.Gupta, A. Akella, S. Seshan, S. Shenkar, and J. Wang, "Undestanding and exploiting network traffic redundancy," UW-Madison,USA,Tech.Rep, 1592,Apr, 2007.

[13] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Watch global, cache local:YouTube network traffic at a campus network—Measurements and implications," inProc. MMCN, 2008, pp. 1–13.

[14] A. Anand, C.Muthukrishnan, A. Akella, and R. Ramjee, "Redundancyin network traffic: Findings and implications," inProc. SIGMETRICS,2009, pp. 37–48.

[15] S. Ihm, K. Park, and V. Pai, "Wide-area network acceleration for thedeveloping world," inProc. USENIX ATC, 2010, pp. 18–18.

[16] Aggarwal, A. Akella, A. Anand, A. Balachandran, P. Chitnis, C.Muthukrishnan, R. Ramjee, and G. Varghese, "EndRE: An end-system redundancy elimination service for enterprises," in Proc. NSDI, 2010, pp. 28–28.

BIOGRAPHY

Syed Sajid Hussain is currently pursuing his master's degree in computer science and engineering at Bharath university, Chennai, India. He completed his B.Tech degree (2009-2013) in the same subject from Islamic University of science and technology Awantipora, Jammu and Kashmir, India. His areas of interest are Neural Networks, Mobile computing, Database Management and Cloud Computing.

AR Arunachalam received his B.E degree in Computer science and engineering from Madras University in 2002 and received his M.Tech degree in Computer science and engineering from Bharath university in 2007.He is currently pursuing his Ph.D in Computer science and engineering at Bharath university, Chennai.He has 10 years of teaching experience and has guided many B.Tech and M.tech projects.

