

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

¹K.Suresh Babu ²K Sekhar

¹Asst.Professor of CSE, Rajiv Gandhi Institute of Management & Science, Sarparam, Kakinada, Andhra Pradesh

²M.Sc student, Rajiv Gandhi Institute of Management & Science, Sarparam, Kakinada, Andhra Pradesh

ABSTRACT:

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.

INTRODUCTION:

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

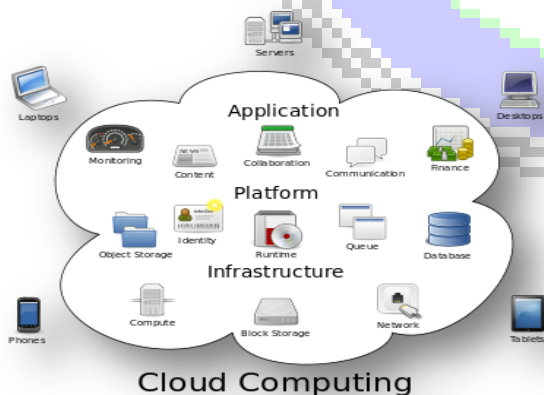
How Cloud Computing Works?

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games. The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

Characteristics and Services Models:

The salient characteristics of cloud computing based on the definitions provided by the National Institute of Standards and Terminology (NIST) are outlined below:

- **On-demand self-service:** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.



- **Broad network access:** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling:** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location-independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.
- **Rapid elasticity:** Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.

Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

IMPLEMENTATION

MODULES:

- Receiver Chunk Store
- Receiver Algorithm
- Sender Algorithm
- Wire Protocol

MODULES DESCRIPTION:

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-1. 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.

Receiver 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.

Sender Algorithm

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

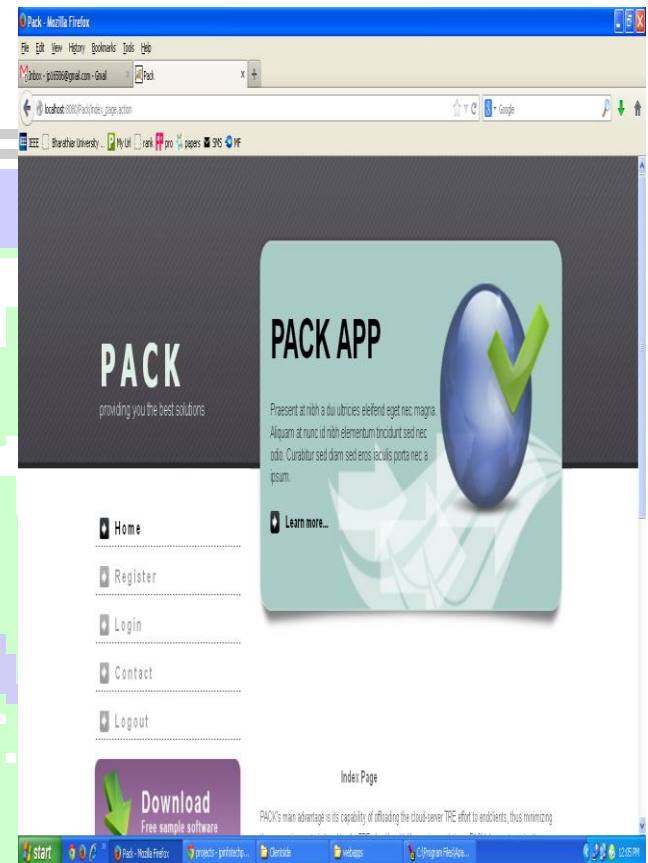
hint. Upon a hint match, the sender calculates the more computationally intensive SHA-1 signature for the predicted data range and compares the result to the signature received in the PRED message. Note that in case the hint does not match, a computationally expansive operation is saved. If the two SHA-1 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.

Wire Protocol

The existing firewalls and minimizes 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.

RESULTS:

SCREEN SHOTS





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 middle-box 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.

REFERENCES

- [1] E. Zohar, I. Cidon, and O. Mokryn, "The power of prediction: Cloud bandwidth and cost reduction," in Proc. SIGCOMM, 2011, pp. 86–97.
- [2] 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.

[3] U. Manber, "Finding similar files in a large file system," in Proc. USENIX Winter Tech. Conf., 1994, pp. 1–10.

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

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

[6] E. Lev-Ran, I. Cidon, and I. Z. Ben-Shaul, "Method and apparatus for reducing network traffic over low bandwidth links," US Patent 7636767, Nov. 2009.

[7] 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.

[8] R. Williams, "Method for partitioning a block of data into subblocks and for storing and communicating such subblocks," US Patent 5990810, Nov. 1999.

[9] Juniper Networks, Sunnyvale, CA, USA, "Application acceleration," 1996 [Online]. Available: <http://www.juniper.net/us/en/products-services/application-acceleration/>

[10] Blue Coat Systems, Sunnyvale, CA, USA, "MACH5," 1996 [Online]. Available: <http://www.bluecoat.com/products/mach5>

[11] Expand Networks, Riverbed Technology, San Francisco, CA, USA, "Application acceleration and WAN optimization," 1998 [Online]. Available: <http://www.expand.com/technology/application-acceleration.aspx>

[12] F5, Seattle, WA, USA, "WAN optimization," 1996 [Online]. Available: <http://www.f5.com/solutions/acceleration/wan-optimization/>

[13] A. Flint, "The next workplace revolution," Nov. 2012 [Online]. Available: <http://m.theatlanticcities.com/jobs-and-economy/2012/11/nextworkplace-revolution/3904/>

- [14] A. Anand, C. Muthukrishnan, A. Akella, and R. Ramjee, "Redundancy in network traffic: Findings and implications," in Proc. SIGMETRICS, 2009, pp. 37–48.
- [15] B. 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.
- [16] "PACK source code," 2011 [Online]. Available: <http://www.evalzo.com/projects/pack>
- [17] A. Anand, A. Gupta, A. Akella, S. Seshan, and S. Shenker, "Packet caches on routers: The implications of universal redundant traffic elimination," in Proc. SIGCOMM, 2008, pp. 219–230.
- [18] A. Anand, V. Sekar, and A. Akella, "SmartRE: An architecture for coordinated network-wide redundancy elimination," in Proc. SIGCOMM, 2009, vol. 39, pp. 87–98.
- [19] A. Gupta, A. Akella, S. Seshan, S. Shenker, and J. Wang, "Understanding and exploiting network traffic redundancy," UW-Madison, Madison, WI, USA, Tech. Rep. 1592, Apr. 2007.
- [20] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Watch global, cache local: YouTube network traffic at a campus network—Measurements and implications," in Proc. MMCN, 2008, pp. 1–13.
- [21] S. Schleimer, D. S. Wilkerson, and A. Aiken, "Winnowing: Local algorithms for document fingerprinting," in Proc. SIGMOD, 2003, pp. 76–85.
- [22] S. Ihm, K. Park, and V. Pai, "Wide-area network acceleration for the developing world," in Proc. USENIX ATC, 2010, pp. 18–18.
- [23] H. Stevens and C. Pettey, "Gartner says cloud computing will be as influential as e-business," Gartner Newsroom, Jun. 26, 2008.
- [24] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "RACS: A case for cloud storage diversity," in Proc. SOCC, 2010, pp. 229–240.
- [25] "Dropbox," 2007 [Online]. Available: <http://www.dropbox.com/>
- [26] E. Allen and C. M. Morris, "Library of Congress and Duracloud launch pilot program using cloud technologies to test perpetual access to digital content," News Release, Jul. 2009.
- [27] D. Hansen, "GMail filesystem over FUSE," [Online]. Available: <http://sr71.net/projects/gmailfs/>
- [28] J. Srinivasan, W. Wei, X. Ma, and T. Yu, "EMFS: Email-based personal cloud storage," in Proc. NAS, 2011, pp. 248–257.
- [29] "Amazon Elastic Compute Cloud (EC2)," [Online]. Available: <http://aws.amazon.com/ec2/>
- [30] "netfilter/iptables project: Libnetfilter_queue," Oct. 2005 [Online]. Available: http://www.netfilter.org/projects/libnetfilter_queue
- [31] A. Z. Broder, "Some applications of Rabin's fingerprinting method," in Sequences II: Methods in Communications, Security, and Computer Science. New York, NY, USA: Springer-Verlag, 1993, pp. 143–152.
- [32] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "TCP selective acknowledgment options," RFC 1881, 1996.
- [33] A. Medina, M. Allman, and S. Floyd, "Measuring the evolution of transport protocols in the internet," Comput. Commun. Rev., vol. 35, no. 2, pp. 37–52, 2005.
- [34] V. Jacobson, R. Braden, and D. Borman, "TCP extensions for high performance," RFC 1323, 1992.