International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 03 | June-2015 www.irjet.net p-ISSN: 2395-0072

Traffic Redundancy and Elimination approach to Reduce cloud Bandwidth and Costs

Ushanandini Balu[1], Resmi. S [2]

[1] *M.tech , Dept. of CSE, Atria Institute of Technology , Bengaluru, Karnataka, India*
[2] *Assistant Professor Dept. of CSE, Atria Institute of Technology , Bengaluru, Karnataka, India*

---***---

Abstract - Cloud computing is expected to trigger high demand for Traffic Redundancy Elimination(TRE) solutions as the amount of data exchanged between the cloud and its users is expected to dramatically increase. We present PACK (Predictive ACKs),a novel end-to-end TRE system, designed for cloud computing customers .The cloud environment redefines the TRE system requirements, making proprietary middle-box solutions inadequate .To reduce bandwidth cost ,Cloud –Based Traffic redundancy elimination(TRE) system should make use of sophisticated use of cloud resources, so that the additional cost of TRE computation and storage can be optimized .This TRE technique uses Predictive ACK's(PACK), designed for cloud computing customers. It gives a methodology to reduce the cloud bandwidth by making use of predictions for the future data, thereby eliminating redundant data. It is a receiver driven TRE technique, that allows the receiver to use newly received chunks to identify previously received chunk chains , that can be used to send predictions for the subsequent data.. This technique does not require the sender to continuously maintain the receiver's status, unlike traditional approach. Predictive ACK's is suitable in pervasive computation environment. It is transparent to all TCP based application and network devices .The main advantage of PACK is that it can offload cloud –server TRE effort to end client, thereby minimizing the processing cost induced by the TRE algorithm.

Key Words: Traffic redundancy elimination, cloud computing, predictions.

1. INTRODUCTION

Cloud providers typically use "*pay* as *you go*" model for their customers. According to the changing needs, the cloud computing customers pay only for the actual use of resources, bandwidth and storage. Depending on the service being used the requirement for bandwidth changes. Communication services like email consumes less bandwidth .Some uncommon services like cloud hosted virtual desktops can place heavy per user-demands on the internet particularly when high resolution desktops or multimedia are deployed. The TRE solution which are currently available requires the server to continuously maintain the status of client. Sender – based TRE add load on the servers ,and is less cost effective. It also requires to maintain end to end synchronization that degrades the TRE efficiency. In most common TRE solutions, the sender and receiver parse the data prior to the transmission . The sender replaces the redundant data with its strong signature[1]-[3].We propose an receiver based end to end TRE technique where the receiver observes the incoming data and tries to match it with the already received chunk chain. If a match is found then the receiver sends predictions to the sender for subsequent data.If redundancy is detected then the sender sends ACK's to the predictions instead of sending raw data .On the receiver side, a new light weight chunking algorithm is proposed called as PACK chunking which is alternative for Rabin fingerprinting.

2. RELATED WORK:

A TRE system called WANAX[1] was developed for the developing world where storage and WAN bandwidth are scarce. For expensive commercial hardware WANAX is a software – based middle-box replacement. The receiver middle box receives the data signatures from the sender middle box. The sender middle sends the signature holding back the TCP stream. The receiver checks whether the data chunks are found in the local cache. If the data chunks are not found in the local cache the data is obtained from the sender middle box or a nearby receiver middle- box. But such a scheme incurs a three way handshake latency for non – cached data.

ENDRE[2] is an end-to-end sender- based TRE for enterprise networks. It uses a new chunking scheme that is faster than the commonly used Rabin fingerprint. The size of the chunks are as small as 32-64 B. The server has to maintains a full and reliable synchronized cache for each client. The size of the cache is kept small , hence the system is not suitable for medium-to-large content or long term redundancy

3 . PROPOSED SYSTEM

The proposed system is a receiver based TRE solution which makes use of predictions to eliminate redundant traffic between cloud and its users. The receiver receives the incoming data stream from the sender. The receiver parses the incoming stream into variable –size , and generates content based signed chunks using SHA-1.

The chunks are compared to the local storage called chunk store .If a matching is found the receiver retrieves a sequence of subsequent chunks called chain .The receiver sends a prediction to the sender for the subsequent data. The prediction include chunks signature and easy to verify hints of the senders future data. The sender verifies the hint and only on the hint match it performs the expensive SHA-1 operation . If redundancy is detected the sender send ACK to the prediction else it sends raw data. The computations are performed by a group of clients instead of cloud distributing the load as the client process only the TRE part. It supports the client to interact with multiple servers and the servers are dynamically relocated around the cloud .It reduces 20% of overall cost to the customer 30% redundancy elimination. It uses the TCP options field and hence all TCP based applications are supported.

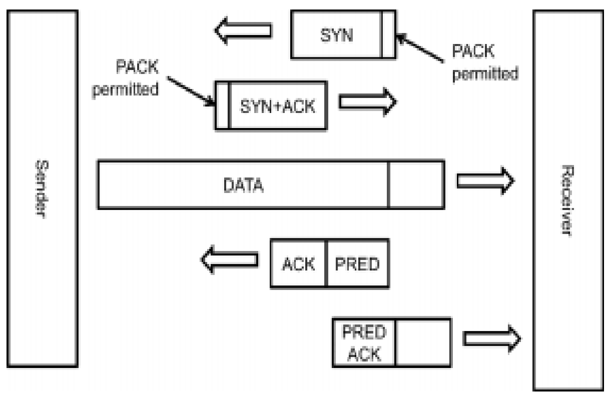

Fig 1 System Architecture for PACK

During the initial TCP handshake PACK is enabled on both the sides. This is done by adding a PACK permitted flag to the TCP options field .PACK assumes that the data is redundant. The sender sends the data in one or more TCP segments and the receiver checks whether the currently received chunk is similar to the chunk in the chunkstore .If the chunks are similar then the receiver sends a TCP ACK message and embeds the prediction in the options field of the packet. If the prediction is true the sender responds with PRED- ACK instead of actual data.

The stream of data received at the PACK receiver is parsed to a sequence of variable-size, content-based signed chunks. The chunks are then compared to the receiver local storage, termed chunk store. If a matching chunk is found in the local chunk store, the receiver retrieves the sequence of subsequent chunks, referred to as a chain, by traversing the sequence of LRU chunk pointers that are included in the chunks' metadata.

Using the constructed chain, the receiver sends a prediction to the sender for the subsequent data. Part of each chunk's prediction, termed a hint, is an easy-to-compute function with a small-enough false-positive value, such as the value of the last byte in the predicted data or a byte-wide XOR checksum of all or selected bytes. The prediction sent by the receiver includes the range of the predicted data, the hint, and the signature of the chunk. The sender identifies the predicted range in its buffered data and verifies the hint for that range. If the result matches the received hint, it continues to perform the more computationally intensive SHA-1 signature operation.

Upon a signature match, the sender sends a confirmation message to the receiver, enabling it to copy the matched data from its local storage.

3.1 Sender algorithm

The sender receives the PRED message from the receiver. On receiving the prediction the sender tries to match the prediction with that of the buffered data. The sender determines the corresponding range for each prediction and verifies the hint .The prediction command consists of chunks signature, an easy to compute function called hint . Only on the hint match the sender calculates the signature using the SHA-1 for the predicted data range and matches it with the signature in the received prediction message. If a match is found then the sender sends an Acknowledgement for the predictions and replaces outgoing buffered data with PRED-ACK. If the predictions are false the sender continues with normal operation.

3.2 Receiver algorithm:

The receiver receives the incoming stream and parses it into chunks. It computes the signature for each chunk and matches it with the local chunk store. If a match is found then the receiver retrieves the chunk from the chunk store and sends a prediction to the sender for subsequent next expected chunks. The prediction carries an offset, and identity of many subsequent chunks. If the prediction is successful the sender responds with PRED-ACK message . The receiver copies the corresponding data from chunk store to its TCP input buffers. Then the receiver sends a normal TCP ACK with the next expected TCP sequence number. If one or more predicted chunks are already sent, the sender resumes with normal operation.

3.3 Receiver chunkstore

A new chain scheme is used by the PACK as shown in Fig 2. This will store the data in the form of chunks. The chunkstore consists of chunks that are linked together according to the last received order .It consists of a large cache of chunks and their associated metadata. The metadata consists of chunks signature and a pointer to the successive chunk in the last received order.

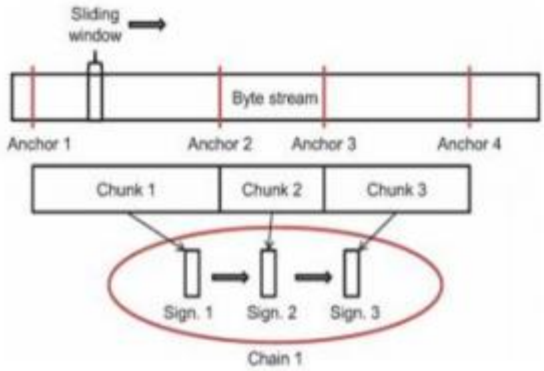

Fig 2. From streams to chain

To retrieve the stored chunk, their signatures and the chain , caching and indexing techniques are employed.

4. CONCLUSION AND FUTURE WORK

This system overcomes the challenges faced by the proprietary middle box solution. PACK is a receiver based end- to – end TRE based on novel speculative principles that reduce latency and cloud operational costs .It is an efficient TRE technique which can eliminate the redundant data based on the power of predictions .Future work may include , if the changes in data are scattered PACK's receiver-based mode is less efficient. So ,when PACK identifies a pattern of scattered changes ,it should drive a sender- receiver approach depending on shared decisions derived from receiver's power or server's cost changes.

REFERENCES

[1]U. Manber, "Finding similar files in a large file system," in Proc. USENIX Winter Tech. Conf., 1994, pp. 1–10.
[2] N. T. Spring and D. Wetherall, "A protocol-independent technique for eliminating redundant network traffic," inProc. SIGCOMM, 2000, vol.30, pp. 87–95.
[3] A. Muthitacharoen, B. Chen, andD. Mazières, "A low-bandwidth network file system," inProc. SOSP, 2001, pp. 174–187.
[4] S. Ihm, K. Park, and V. Pai, "Wide-area network acceleration for the developing world," inProc. USENIX ATC, 2010, pp. 18–18
[5] B. Aggarwal, A. Akella, A. Anand, A. Balachandran, P. Chitnis, C. Muthukrishnan, R. Ramjee, and G. Varghese, "EndRE: An end-systemredundancy elimination service for enterprises," inProc. NSDI, 2010,pp. 28–28.
[6]E. Zohar, I. Cidon, and O. Mokryn, "The power of prediction: Cloud bandwidth and cost reduction," inProc. SIGCOMM, 2011, pp. 86–97.

BIOGRAPHIES

Ms Ushanandini Balu has completed B.E in Computer Science at Visvesvarayaechnological University, India in 2009. She is currently pursuing M.Tech in Computer Science from Visvesvaraya Technological University, India. She is interested with areas of research related to Cloud Computing, Computer networks

Mrs. Resmi.S has completed B.E (CSE) from Calicut University and M.Tech (CSE) Visvesvaraya Technological University. She is currently working with Atria Institute of Technology, Bangalore as Asst Professor. She is interested in Compiler Design and Analysis of algorithms.

