Chord Protocol Algorithm

In today's Internet, Peer-to-Peer architecture are becoming the most successful means for sharing massive computation and storages, which cannot be achieved using Client-Server Model. In peer-to-peer networking, every node has the capability of being converted into a client and a server node, depending upon the current requirement of the whole network .Hence these systems are scalable and do not have single point of failures. But these suffer from various drawbacks such as redundant storage, selection of nearby server or nodes, search according to hierarchical naming convention, locating data items. In order to overcome those obstacles, researchers found structured P2P overlay networks where contents are placed at specific location to make frequent data requests using Distributed Hash Table .Structured P2p overlay networks are Chord, Content Addressable Network(CAN) ,Tapestry, Pastry ,Viceroy and Kademlia and all of them allows efficient way of locating data items . In this research paper, I will be discussing one of the famous scalable and efficient algorithm in P2P distributed hash table, known as CHORD. It was developed by group of members at MIT.

Chord is a Peer to Peer DHT protocol for Web applications that solves problem of finding a data item in large number of distributed computers (known as "forks" or "nodes"). It works similar to P2P distributed Hash-table. DHT consist of storing keys and corresponding values of different forks. Chord protocol has the functionality of describing how mapping of the key to the specific fork are achieved, and how forks can discover other forks by first locating the key value assigned to it. The five main advantages of using Chord in P2P file system are:

a) **Load Balancing:** It provide load balancing by using Secure Hashing Algorithm SHA-1 ,which basically spread keys over the network nodes
b) **Scalability:** It provides routing information for N nodes in O (log M) steps, then rectify lookups using O (log M) signaling messages to other network nodes. Hence efficient for large systems.
c) **Decentralization:** It provides correctness, performance and robustness even when a network node join or leave the network, or in case of node failures.
d) **Flexibility:** It have a structured lookup algorithm that can resolve messages to other nodes (recursively or iteratively). This gives flexibility in mapping the unique names to Chord keys.

e) **Availability:** It adjust it successor-list/internal table to reflect nodes that had newly joined as well nodes leaving (called as node failures), ensuring continuous modification in network.

Let's discuss about Chord Protocol. How it is implemented using 1-dimensional ring? What is methodology of implementing DHT in Chord algorithm? When finger tables are used and how they are used?

Chord protocol consist of arranging network nodes in a circle based on the node identifiers as shown in Fig [1] (chord protocol Ring). Generic Keys need to be assigned to all the successor nodes in the network. Let us consider there are M Network Nodes and K Keys, then each nodes and keys are allocated to p-bit unique qualifier using Rendezvous hashing or Consistent Hashing algorithm, known as Secure Hash algorithm or SHA-1. This hash function is used for evenly distributing nodes and keys on the circle. Hence, the limit of possible number of hashes for DHT lies in range of 0 and (2^p-1) inclusive.

Each node on the circle, also called as Peer has its identification stored in format of (Key, Value). Whenever a peer request resources, we perform a single operation of assigning keys to the Chord node. The protocol consist of generating keys, which form the base of finding location of data item or node in circular overlay. The hashing function when applied to IP address of a node produce Node identifiers, whereas when hashing function is applied to unique Node name's, it produces Node (or data item's) Key. Chord consist of ordering nodes, where successor node in each network is the one which has the next greatest hash value and will be placed in a clockwise rotation. It maintain efficient uniform distribution of keys using virtual nodes. In virtual node, multiple ID's are assigned to physical node in the same Chord System.

Finger Table: stores routing information records and beneficial for optimized search operation. Each node maintain a finger table to record the log M successors (M is number of neighbors or peers).The table consist of "n entries" where n is number of bits used to represent hash in key-space of Distributed Hash Table (e.g. 128).The kth entry (0< K < n), is of the form of hash of h+2^k (h is hash of current node). Every peer checks its predecessor and successor after regular intervals and updated the Finger Table. Whenever a new peer is added to the circular ring, it contacts a peer predecessor already in the network and updates its entry into the finger table. Similar operation happens when peer leaves the network. Chord routes the message to the

destination node or identifier, by requesting successor and then forwarding packet to the successor node that is nearest to destination. The total cost of the network remains ½ log M where M is number of overlay peers.

Chord protocol consist of iterative or recursive method of implementation. In iterative method, lookups are performed to a sequence of nodes consisting of querying their finger tables and moving nearer to successor on circular ring. In case of recursive method, the adjacent nodes forward request to next node until the successor node is reached.

APPLICATIONS OF CHORD PROTOCOL

Cooperative Mirroring: In cooperative mirroring, the chord protocol implements balancing of load for distributed information across all the peers/nodes. Example would be a group of Software developers working together to publish a distribution, which very according to newest release and relatively unpopular release. In this case developers need to mirror each other's distribution in order to have load balancing across all servers, ensuring authentication .This creates a reliable decentralized system that has no central authority and data is replicated and cached across multiple distribution (distributed data storage) as shown in fig 2.

Time-shared storage: In this application, chord protocol manage availability of data when nodes are intermittently connected. If a person wish to access his data all the time but he is online rarely, then he can use similar technique to bartering system. He can offer his services to store other's data when he has fully operational online connection and in return earn the benefit of storing his data when the he has server down. This increases availability of data.

Distributed indices: In this application, chord protocol is used for retrieval of files within searchable database. It is used in unstructured P2P overlay network and have Keyword-based search which results in list of machines providing files or documents related to keyword.

Chord-Based DNS: Domain name server provide a lookup service for mapping IP Address to domain name and vice versa. Chord protocol can provide this service by using (key, values) where keys are the host names and values are their IP addresses. It hashes the host name to key .Chord has no root servers, no naming structure, no manual management of routing information and can easily find objects not tied to specific machines. On the other hand, DNS requires root servers, naming structure and hence more overhead as compared to Chord.

POSSIBLE ATTACKS AND COUNTER-MEASURES USING CHORD PROTOCOL

Denial of Service Attack: The files placed in distributed storage are susceptible to target-file attack, where a small set of files are attacked repeatedly.

The researcher's solution is "Location Guard"- which consist of three components as shown in fig 4. First, location key that defines location of a file based on key index and consist of a randomly arranged string of bits (128 bits).Second, routing guard is an invulnerable algorithm that prevents unauthorized access to documents or files in an overlay network. Third, a group of inference guards for location. The file content are protected from an adversary by implementing cryptographic mechanisms. Location guard provides techniques to hide the documents in structured network such that only an authorized person possessing key for file's location will be able to find the file effectively. Location Guard protects target document from host compromise attack, location interference attack and DoS attack with minimal storage overhead and performance.

Byzantine Attacks: It is an attack model in which attacker allows many Byzantine peers to join into the circular overlay network. The attacker carefully selects the IP-addresses of peers so that they are placed at critical locations.

The researcher's solution is "S-Chord" which enforce strict rules on peers joining the network and prevent their undesirable behavior. Example would be -After every 25 searches that a peer performs on the data, the peer must provide services to any one search request otherwise they will be disconnected from the network.

CONCLUSION

In this research paper, I have summarized one of the popular structured P2P overlay network called as Chord Protocol. Chord Algorithm is systematic lookup algorithm for distributed systems. The structured P2P overlay increases the performance in Wireless mesh Network (WMN) which have high bandwidth. The key terminologies of the chord protocol are DHT, finger table, successor's, node key, node's value, circular ring, iterative and recursive chord implementation. Then we discussed four application of Chord Protocol i.e. Cooperative Mirroring, Time-shared storage, distributed indices, and Chord-Based DNS. Finally, I outlined the various strategies that are adopted by attackers to attack target files in DHT and their possible counter-measures that have been developed in past. Some of the counter measures are Location guard to prevent DoS attack

and S-Chord to prevent Byzantine Attack. In future, techniques such as key consistency and data consistency can be introduces in order to have much more effective lookup operation implemented under a timing constraint with best probability distribution of keys.

REFRENCES

1. P. Prasanna Murali Krishna and M. V. Subramanyam, "Chord Protocol Investigation in P2P WMN-Wireless Mesh Network with Mobility," (WAS abbreviated as *World Academy of Science, International Journal of Electrical Engineering and Technology IJEET, Computer, Energetic, Electronic and Communication Engineering* Volume: 9, Issue:9), Year 2015.
2. Ding Zhi-min and Qian Quan ,"Load Balancing Algorithm in Chord-Based P2P network" (*Computer Science Department, Shanghai University, Shanghai, China),* 2014.
3. Stoica and R Morris "Chord: Scalable P2P look-up service for Internet Applications", (*Association for Computing Machinery's Special Interest Group on Data Communication SIGCOMM),* Volume: 32, Issue: 1, no A, On Page(s): 149- 160, (2002).
4. Mudhakar Srivatsa and Ling Liu," Mitigating DoS Attacks on the Chord Overlay Network", (*Institute of Electrical and Electronics Engineers- Transactions On Distributed and Parallel Systems*, Volume: 20, Issue: 4), April 2009.

BIBLIOGRAPHY

Krishna, Subramanyam and K. Satya Prasad, "Chord Protocol Investigation in P2P WMN-Wireless Mesh Network with Mobility," (WAS abbreviated as *World Academy of Science, International Journal of Electrical Engineering and Technology IJEET, Computer, Energetic, Electronic and Communication Engineering,* Volume: 9, Issue:9), Year 2015.

Stoica, R Morris and D Karger " Chord: scalable P2P lookup service for internet applications", (*Association for Computing Machinery's ACM Special Interest Group on Data Communication SIGCOMM)),* Volume: 32, Issue: 1, no A, On Page(s): 149- 160, (2002).

Figures

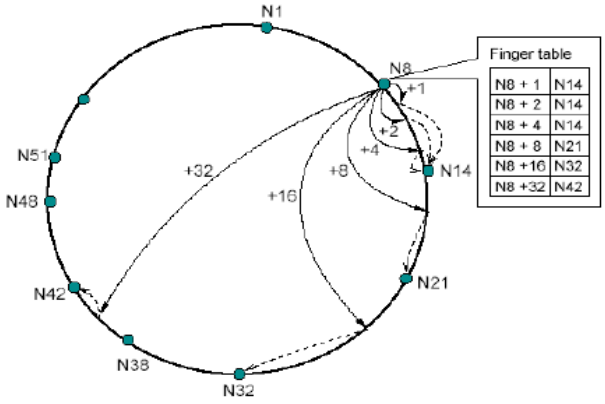

Fig. 1 The CHORD Protocol Ring

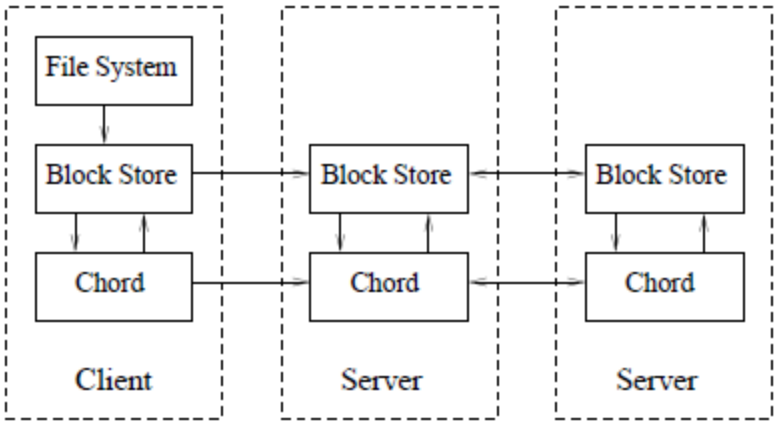

Figure 2: Structure of an example Chord-based distributed storage system.

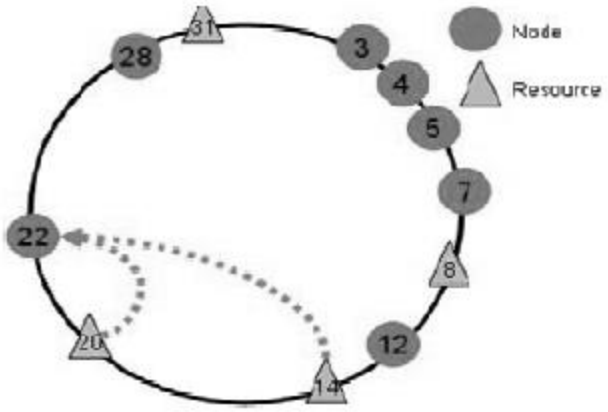

Fig.3 Chord Overlay with Peers and Resources

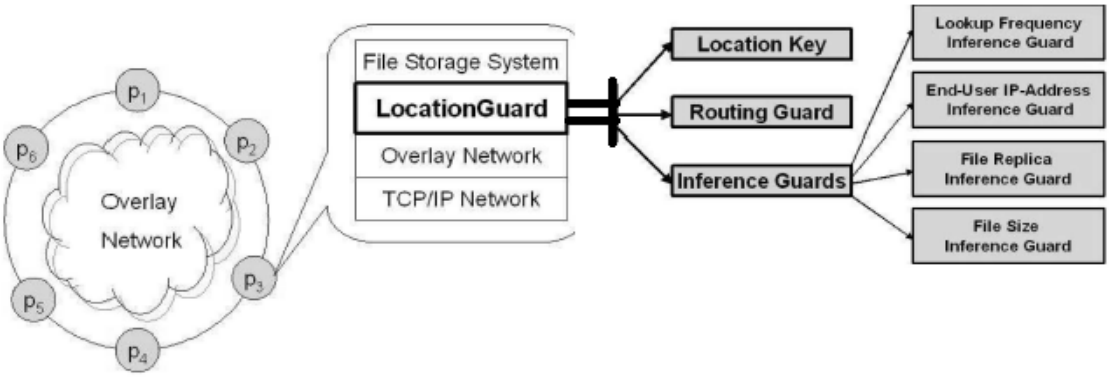

Fig 4 Location Guard: System Architecture