IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 2, February 2018

Adaptive and Co-operative Caching System for Efficient Bandwidth Utilization in MANET

Selvaraj D[1], Arulkumar D[2], Dhinakaran D[3]

Professor, Department of Electronics and Communication Engineering, Panimalar Engineering College, Poonamallee, Chennai, India[1]

Assistant Professor (G-I), Department of Electronics and Communication Engineering, Panimalar Institute of Technology, Poonamallee, Chennai, India[2]

Assistant Professor, Department of Computer Science and Engineering, Peri Institute of Technology, Manivakkam, Chennai, India[3]

Abstract: This paper presents the co-operative based database for ad hoc Mobile Networks (MANET). The main focus of the system is on the nodes that save the queries sent. Queries are used as an index of the data stored in the caches of the nodes. This paper deals about the system formation and detection of demanded data from the cache or from the external database. Different parameters like upper and lower bonds for query caching nodes, average load experience, node bandwidth and hit ratio are analyzed and derived. The proposed method is simulated using NS-2. The proposed system has achieved a better success rate and a lower delay compared to other caching systems.

Keywords: Mobile ad hoc networks, distributed cache, cache indexing, database queries.

I. Introduction

The challenges in MANETs leads to the creation of efficient routing techniques [1]. In many scenarios, mobile devices can be distributed over a large area where access to peripheral data is achieved through one or more Access Points (AP). However, all the nodes don't have direct contact with the access points, but depend on other nodes that act as routers. In some situations, the access points can be located at the ends of the MANET, where reaching them can be expensive in terms of delay, power consumption and bandwidth. This proposed framework creates a cooperative cache system that reduces the delay and maximizes the probability of finding saved data in the ad hoc network, without inducing exceptionally high node traffic in several node cases at an average speed (at high traffic and low traffic conditions) and show how our functions conclude the rest [2].

II. Introduction Existing System

A preface system proposed in [3] to store the response of the database to queries in given nodes and use queries as a directory for responses. Parallel to this, Three associated cache schemes: Cache Path, Cache Data, and Hybrid Cache is also covered. The main idea of these schemas is to analyze the step requests and data cache or the address of the node where they are stored. Later, if the same data request goes through the node, you can present the data or redirect the request to its stored location.

The cache path saves space by storing locations where data is to be stored, while cache data saves time by storing data instead of the route. The third scheme, Hybrid Cache, is an intermediate solution where queries are cached by route or data, whichever is optimal [4] The relative positions of the nodes with each other and with the AP are very essential to the success of the queries. previous systems. On the other hand, in [5], the cache software called SQUIRREL has been integrated into the Internet nodes to allow multiple nodes in a given region to share their caches..

III. Proposed System

An algorithm for a system that would outperform the Data discovery under ICP (Internet Cache Protocol) at various scenarios which would show a better performance metrics in terms of delay, jitter, network traffic (under low as well high traffic condition) effective bandwidth utilization and good put is proposed in this paper. The competence of query directories (QDs) can be improved by using query compression techniques similar to the one proposed in [6], which may prove helpful in large networks.

IV. Introduction Algorithm Used

COACS Algorithm:

i. Initialize all the nodes.
ii. Some nodes are elected as Query Directories (QD) using QDAP (Query Directory Assignment Packet) based on the score calculated { Time, Battery, Memory and Bandwidth }

DOI 10.17148/IARJSET.2018.5211