Performance of Enhanced Distance Vector Multipath Routing

Kanwalinder Jit Kaur
Lecturer
Department of Computer Science, Punjabi University, Patiala, India
Tel.:+91-9815612289(Cell), +91-175-2286224(R), +91-175-3046312(O)
Fax: +91-175-3046313
E-mail: kanwal_gheek1@yahoo.com
Dr. Jyotsna Sengupta
Reader
Department of Computer Science, Punjabi University, Patiala, India
Tel.:+91-9316201605(Cell), +91-175-2215251(R), +91-175-3046312(O)
Fax: +91-175-3046313
E-mail: jsengupta1@lycos.com

Abstract-In this paper, a new algorithm named Enhanced Distance Vector routing algorithm has been developed to make the existing distance vector routing secure. This algorithm has been extended to multipath routing to make multipath data transmission secure. The performance of this algorithm has been found out by using simulation environment of *ns-2*. The results show that multipath Enhanced Distance Vector performs better than single path Enhanced Distance Vector in terms of throughput by 10 times better and cumulative distribution better by about 46%.

1. INTRODUCTION

Distance vector routing algorithms operate by having each router maintain a table, that is, a vector, giving the best known distance to each destination and which path to use to get there [7]. These tables are updated by exchanging information with the neighbors. Each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. These entries are preferred outgoing line to use for that destination, and an estimate of the time or distance to that destination. The metric used might be number of hops, time delay, and total number of packets queued along the path.

For enhancing the security, the most obvious solution is to use a complex encrypting algorithm. It is common for present day attackers to check a traffic stream of a desired IP source/destination pair and record the transaction data, then apply sophisticated tools to decode the information. Although this is a time consuming complex operation, with the right deciphering software, there is a possibility that the file might be decodable, thus releasing confidential/secured information [3]. Hence, the use of a complex encryption tool alone will not be the complete solution. Therefore, coupling the encryption process with a multipath routing topology will increase the security of data. Traditional protocols like the link state protocol [2] and the shortest path protocol [2] will provide data communication through the shortest path, which commonly is a fixed single path that does not change, considering the stability of the network connections and switches/routers. If the path from source to destination is monitored, there are many intermediate nodes. An attacker can try to hack into any one of these intermediate nodes. But in the proposed topology of multipath Enhanced Distance Vector, the path is never fixed and hence this becomes an effective way of protecting the network from attackers. Even if an attacker gains access and copies the data at a particular node, it will not be of significance because only a part of the encrypted message is obtained.

Ad hoc On Demand Distance Vector Routing (AODV) [6] algorithm has been proposed for mobile nodes. These nodes do not require intervention of any central point or existing infrastructure. Ad hoc on-demand multipath distance vector (AOMDV) [5] algorithm is a multipath extension to ad hoc on-demand distance vector (AODV), which is a single path routing protocol. AOMDV computes multiple loop-free and link-disjoint paths. In mobile ad hoc networks, one might expect multipath routing to provide some robustness to link failures and facilitate the transmission of packets along paths that avoid regions of congestion. TCP has been considered as goodput as the metric of performance to improve network performance in terms of the achieved throughput [9]. The performance issues of destination-sequenced distance vector (DSDV) and ad-hoc on-demand distance vector (AODV) routing protocols for mobile ad hoc networks have been measured [4]. The issue has been discussed for mobile hosts but more or less the same has not been discussed for static hosts, where performance of multipath algorithm is out performing than that of single path.

II. ALGORITHM DEVELOPED

Enhanced Distance Vector (EDV) routing algorithm has been implemented in two ways, single path and multipath. The multiple paths chosen for data transmission are of equal

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 549–552.

cost. Analysis has been done on the working and efficiency of the algorithm on these two different scenarios.

Enhanced Distance Vector Routing is the implementation of Distributed Bellman Ford (or Distance Vector) routing. The implementation sends periodic route updates every *advertInterval*. This variable is a class variable. Its default value is 2 seconds. In addition to periodic updates, each agent also sends triggered updates, it does this whenever the forwarding tables in the node change. This occurs either due to changes in the topology, or because an agent at the node received a route update, and recomputed and installed new routes. Each agent employs the split horizon with poisoned reverse mechanisms to advertise its routes to adjacent peers. "Split horizon" is the mechanism by which an agent will not advertise the routes to a destination out of the interface that it is using to reach that destination. In a "Split horizon with poisoned reverse" mechanism, the agent will advertise that route out of that interface with a metric of infinity. Each DV agent uses a default preference_ of 120. The value is determined by the class variable of the same name. Each agent uses the class variable Infinity (set at 32) to determine the validity of a route. Because distance vector routing works in theory but has a serious drawback in practice that it converges to the correct path, it may do that slowly. Moreover, it reacts rapidly to good news, but leisurely to bad news. To overcome this, count to infinity, problem variable Infinity has been set.

III. SIMULATION ENVIRONMENT

NS-2 [1, 8] has been used for simulation study. The hosts are placed on a square field of 1000m x 1000m. The constant bit rate (CBR) traffic is used in the simulation. Each connection is specified as a randomly chosen source-destination (S-D) pair. The packet size is fixed as 512 bytes. The packet sending rate is 4 packets per second. Each connection starts at a time randomly chosen from 0 to 100 seconds. Simulations are run for 8 simulated seconds. Each data point represents an average of ten runs with identical traffic model.

IV. PERFORMANCE ANALYSIS

The performance of the algorithms has been evaluated using ns-2 simulator. The following key issues have been addressed:

1. Throughput of generating packets.
2. Throughput of sending bits vs minimal simulation End to End delays.
3. Throughput of sending bits vs average simulation End to End delays.
4. End to End Simulation Delays vs Cumulative Distribution.

Results:

1. Throughput of generating packets: The number of packets generated in multipath is 10 times higher than number of packets generated in single path with in same simulation time, where packet generating pattern is almost the same.

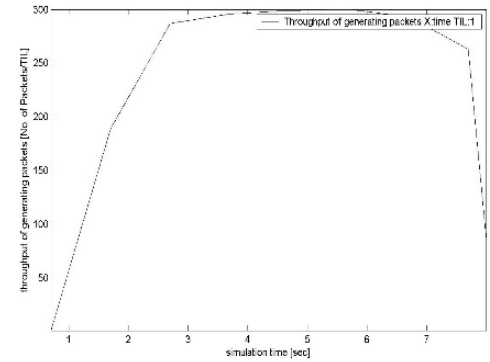

Fig.1 Throughput of generated packets for Single Path Enhanced Distance Vector

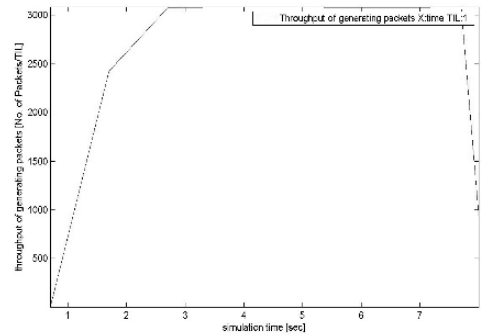

Fig.2 Throughput of generated packets for Multipath Enhanced Distance Vector

2. Throughput of sending bits vs minimal simulation End to End delays: In multipath throughput of sending bits with respect to minimal simulation End to End delay shows 10 times better results than single path.

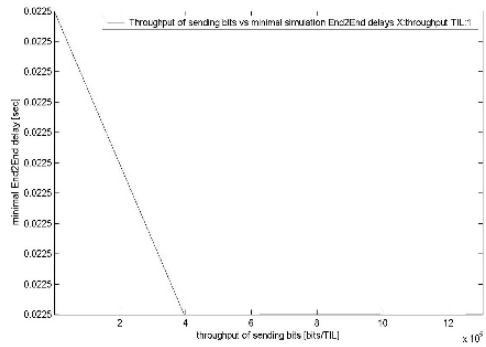

Fig. 3 Throughput of sending bits vs minimal simulation End to End delays for Single Path Enhanced Distance Vector