Enhanced Reinforcement Learning Based Privacy Preserving Circuit Construction and Routing in an Onion Routing Network

S. Shakila, and Gopinath Ganapathy

***Abstract*—**Onion Routing is the process of providing anonymity to a packet by rerouting it through various intermediate routers in an encrypted format. Hence the major requirement of an onion routing network seems to be security. One of the other major hurdles that is overlooked is the slow nature of the transmission when an onion routing network is used. The current paper provides an effective solution that not only provides secure transmission, but also construct routes that can perform faster transmissions. Reinforcement Learning method is used to provide adequate exploration and exploitation during the path selection process. It also uses user defined QoS parameters and their corresponding weightage values as a base for router selection. Probability based router selection is carried out along with an exploration functionality, that maintains diversity in the obtained solutions.

***Keywords*—**AHP, Circuit Construction, Cumulative Distribution Function, Onion Routing, Reinforcement Learning, Weighted Sum method.

I. Introduction

Anonymity is the property of being unidentifiable within a group. Several protocols have been developed that allow one to communicate anonymously over the internet. The Tor protocol shows high degree of anonymity than other protocols. A lot of research has been carried out to measure anonymity and to guard against potential attacks. Due to the wide use of Internet based applications, Hyper Text Transfer Protocol(HTTP) traffic comprises an overwhelming majority of the connections and it is unclear whether TOR can facilitate interactive web browsing[18]. The Selection of appropriate routes is one of the major functionalities of a TOR network. The complication of this mechanism is that it also requires two other basic functionalities for successful operation; the generated route should be the fastest and the most unpredictable. Achieving both these functionalities to the fullest is not feasible. Further, speed has always been a trade off for the need of security. Hence a TOR network always remains a slow and secure routing structure. Any packet that is passed through a TOR network has always been found to reach the destination taking at least 3x or 4x times of the transmission time taken by a normal transmission. Practical usage delays are not mostly tolerated by end users. Hence this proves as a serious downside, which discourages users from using the onion routing system.

The remainder of this paper is structured as follows; Section II presents the related works, Section III explains our approach of secure and fast circuit construction, Section IV presents the results and discusses them and Section V concludes the study.

S. Shakila, Government Arts College, India
Gopinath Ganapathy, Bharathidasan University, India

II. Related Work

A trust based routing methodology for onion networks that guards specifically against interference attacks has been presented in [8]. The problems in conventional routing methods have always been the fact that if the intruders have prior knowledge about the trust degrees present in the system, then anonymity becomes compromised. Hence the paper [8] provides a trust degree based methodology, that helps defeat the interference attacks. A similar trust based approach that uses trust graphs is proposed in [14]. Interference attack is the major attack being carried out on any system using trust based communications. This attack has been thwarted by [14] using restricted user knowledge. It uses three unique properties for performing routing namely; group trust is maintained, that verifies the trust levels assigned by users, an adaptive trust propagation system is maintained, which derives the global trust from the trust graphs and it works on a completely decentralized environment. Paper [15] uses a similar latency graph based privacy preserving mechanism that exhibits resilience link based attacks. It also aims to reduce the delays caused in the onion routing system. A similar paper that aims at reducing the delays in an onion routing system by measuring the latencies and other parameters is described in [16].

A forward secrecy based non-interactive onion routing approach was proposed in [9]. This method achieves its required functionality in a fully non-interactive manner without requiring communications from the router or users. In addition to this, [9] also provides faster key management with a comparable computation cost. The forward secure public key proposed in [10] was utilized in [9]. Identity based forward secure encryption is provided by applying the generic paradigm proposed by [11] to the Hierarchical Identity based encryption approach [12].

A provably secure and practical onion routing method is

proposed in [13]. Analysis has been carried out in the current protocol deployed in onion networks and it was observed that the current protocol still lacks security guarantees that are the major requirements of the next generation onion routing networks. Paper [13] provides the properties required for an effective OR system.

III. Reinforcement Learning Based Privacy Preserving Circuit Construction and Routing in an OR Network

Security and faster transmissions have been the mandatory components in a TOR network. Security has been given major thrust, while the speed of data transmission has always been assigned lower importance. The major downside in a TOR network that had been often felt is high transmission time. Due to the additional overhead of increased packet transmission and encryption, the time taken by any packet transmitted through a TOR network always increases. The process of providing a mechanism that reduces the time taken for transmission through a TOR network is simple, but problem arises when security is lowered as a tradeoff for time. This becomes unacceptable. Considering the QoS parameters related to transmission while discovering a route, provides an effective solution for determining the best route, and by incorporating the information about network traffic, the route that has been selected can be considered much more reliable. This scheme is used as the basic logic for route determination by the authors.

Three basic components of a TOR network are the entry nodes, exit nodes and other nodes. Entry nodes perform the most basic and the most important task in a TOR network; determining the level of encryption. Packets sent through a TOR network requires different levels of encryption based on their importance. The process of encrypting the packets depending on the requirement is performed by the entry level nodes. These nodes receive packets in their true form without any encryptions; hence the entry nodes are high performance and highly secure nodes that cannot be compromised easily. The exit nodes are the ones that strip off the final layer of encryption, hence they are also made reliable and secure. The remaining nodes perform the intermediary process of stripping off the encryption layers and forwarding the packet to the next node.

The encryption level for a packet is determined by the application transmitting the packet [1]. Higher level of encryptions are provided to throughput sensitive applications, while encryption levels are reduced as the transmission type comes down to delay sensitive applications. This mechanism is clearly explained in the previous work of the authors [1].

The method of secure route selection begins during the initial setup of the TOR network. The components (nodes) of the network and their distances are recorded initially with respect to every router. A TOR system in general does not perform route selection; instead, it performs the process of next node selection. This mechanism eliminates the possibility of backtracking. The routing algorithm is distributed to all the routers in the network.

Algorithm:

1. Setup the network nodes(router) and construct the graph

2. Find the neighbor set N for each node

3. For every packet encountered, perform the following

a. If the current node is not an exit node then

i. Calculate the weighted sum of each node in the network using success rate, failure rate and other QoS parameters using (2)

ii. Determine the probability of selection of each node using (3)

iii. Find the destination router D_r using CDF

iv. Repeat (iii) until the termination condition is reached (exploration depletion or D_r not in EL)

v. Add D_r to EL if it is not in EL

b. If the current node is the exit node, then strip off the final layer of encryption and forward the packet to the destination

When a packet is encountered, it triggers the process of route selection in a node (after adding encryption layers, if the node is an entry node). According to this method, determining a route not only depends upon the distance between the current node and the destination node, it also depends on the QoS parameters related to transmission. The quality parameters used in this paper are jitter, delay and the level of network traffic currently encountered in the network edge. Further, the system also considers the success and failure rates of the current node under examination. Each of these parameters is provided weights ranging from -10 to 10. Since different networks are constructed with different functionalities in mind, a hard coded value will not be an appropriate option. Hence this process is performed by the network administrator before deployment. All the routers in the network work with the same parameter weights, hence this is a onetime process. The method of Analytic Hierarchy Processing (AHP) is used to perform weight assignments. This can be performed using pairwise comparison [2] or using direct weight assignments [2].

In pairwise comparison, each pair of parameters is provided to the user and the ranking is provided with respect to each other. This method can be used if the number of parameters is very large or if the user is unsure about the ratings of parameters in the individual sense. The comparison matrix depicting the weights set W is as follows.

$$W = \begin{bmatrix} w_1/w_1 & w_1/w_2 & \cdots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & \cdots & w_2/w_n \\ \vdots & \vdots & \ddots & \vdots \\ w_n/w_1 & w_n/w_2 & \cdots & w_n/w_n \end{bmatrix} = \begin{bmatrix} 1 & s_{12} & \cdots & s_{1n} \\ s_{21} & 1 & \cdots & s_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ S_{n1} & s_{n2} & \cdots & 1 \end{bmatrix} \quad (1)$$

Where W_x/W_y compares the parameters x with respect to y. S_{xy} represents the comparison score of x when compared with

y.

In direct user assigned weights, as the name implies, the user assigns weight values for attributes directly.

After the assignment of weights and the quality requirements for parameters, the weighted sum method [3,4] is used to calculate the importance value of each router (WS_r)

$$WS_r = \sum_{i=1}^{x} W_i * N_{ri} \quad (2)$$

Where Wi represents the weight of the property *i* and N_{ri} represents the normalized value of the i^{th} attribute for the router r. Normalization is performed on the attribute to make certain that irregularities are removed and operations in certain values do not cause huge deviations in the result. The values for parameters obtained from the router are normalized in the range 0.1 to 1. 0 is avoided such that even if a value is the lowest in its scale, it will also have some influence on the result. The only factor that can influence the outcome (result) to a large extent are the weight values.

The next step involves in finding the probability of routers to facilitate the selection process.

$$P_r = \frac{WS_r}{\sum_{j=1}^{n} WS_j} \quad (3)$$

The probability of selecting a router Pr is determined by dividing the weighted sum of the router WSr by the sum of all the weighted sum values.

The selection process is carried out using the Cumulative Distribution Function (CDF) [5].

The cumulative distribution function of a real-valued random variable X is the function given by

$$F_X(x) = P(X \le x), \quad (4)$$

When the right-hand side represents the probability that the random variable X takes on a value less than or equal to x. The probability that X lies in the semi-closed interval(a,b), where a<b, is therefore

$$P(a < X \le b) = F_X(b) - F_X(a). \quad (5)$$

Exploration and exploitation are the major components of any heuristic based search algorithm. Exploitation is performed by using the already identified best solutions, while exploration is performed to determine better or more optimal solutions from the system. A Tabu List [6,7] is used by the authors to control the exploration and exploitation phases, which will be called as the Exploitation List (EL). We do not use the name Tabu List due to the fact that a few modifications have been performed in the usage scenario of the list. The size of the EL is maintained with a maximum limit of 10% of the network size.

After the process of selecting a destination router (Dr) using the CDF, the EL is checked for Dr in its list. If Dr is not present in the list, then it is added to the list else the process is repeated until a value that is not in the EL is determined or until exploration depletion (ε), where ε=size of EL. The value currently contained in Dr is taken as the next hop.

IV. Results and Discussion

Analysis was carried out using 75 nodes. The network is maintained as a complete graph i.e. all the nodes are connected to all the other nodes, in order to increase the destination choices and to determine the system's behavior during exploration and exploitation phases. 10595 packets were transferred over the network and the node that has been selected for transmission, transmission result (successful or retransmitted) and the time taken for transmission were recorded. The properties used for analysis in the current simulation are success rate of the router, failure rate of the router, network traffic, bandwidth, jitter and delay. Every router is made to maintain the success and failure rates of all its neighbors, along with the bandwidth details. Network traffic is computed dynamically during transmission and delay is calculated using the difference between the packet sending and packet receiving time.

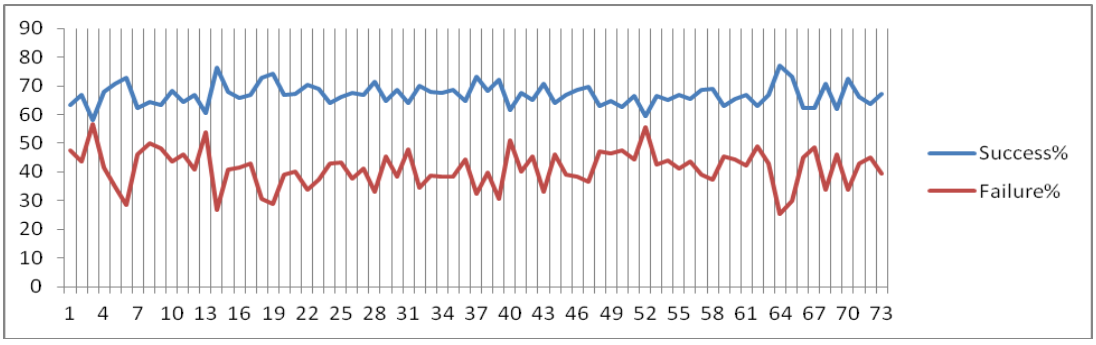

Fig. 1 Success and Failure rates of the nodes

The success and failure rates of the nodes in the network are depicted in Fig 1. It can be observed that the success rate graph is always above the failure graph, which proves that the deployed system performs efficiently in terms of selection, by maintaining low failure rates.

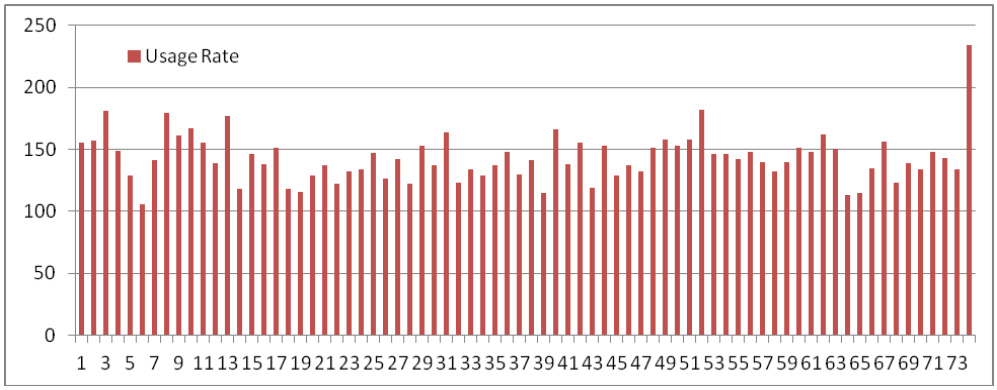

Fig. 2: Node Usage Graph

Node usage during the simulation process is depicted in Fig 2. It can be observed that on an average all nodes have similar usage patterns. This explains the efficient functioning of the exploration and exploitation modules. Very few best nodes are available in the network that are exploited, while all the other nodes are used on an average scale.

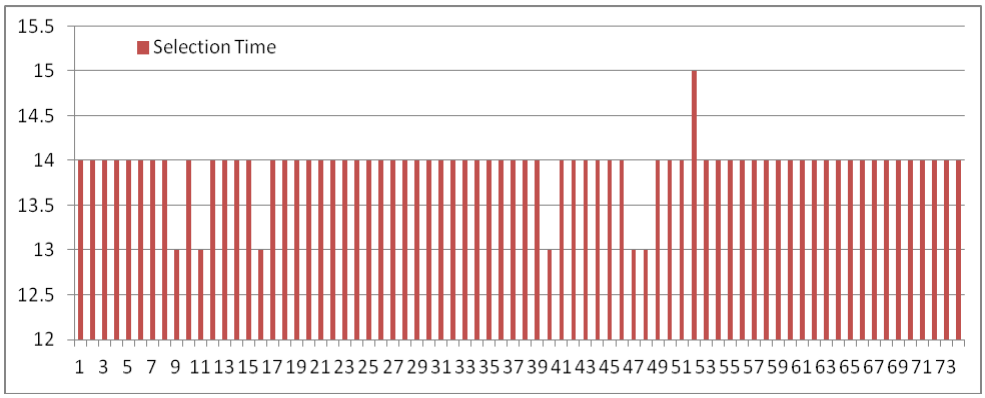

Fig. 3: Time Taken for Node Selection

Time taken during the selection process is depicted in Fig 3. Since each node has been used multiple times during the transmission process, the average time taken for each node is considered for plotting. It can be inferred from the figure that time taken for 95% of the available nodes is the same. It can also be observed that the time taken for the nodes with highest success rates is lesser than the others.

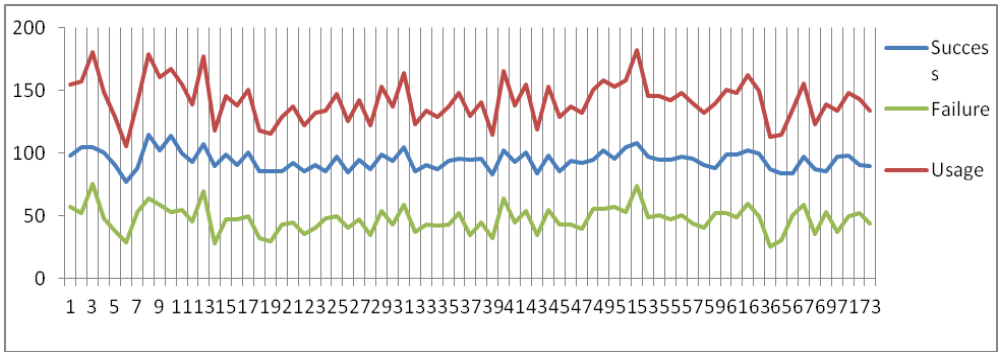

Fig. 4: Success, Failure and Usage Comparison

Fig. 4 depicts the comparison between the success, failure and usage of the nodes. It can be observed that the usage graph indicates proportionality to the success and failure graphs. As the success rate increases, the usage level also increases and vice versa.

V. Conclusion

The paper presents a novel Reinforcement Learning method that uses probability and a variant of Tabu List (EL) to carry out the learning process. The process is to be carried out by every router in the network, hence the complexity of the algorithm is kept to the minimum *O(n)*. This value is

actually due to the fact that our network considers a complete graph. This value would actually be the maximum degree of the router being used. Usage of EL provides the system with sufficient bounds for exploration. Further, flexibility is provided to the user by permitting custom parameter set and weight assignment modules. Due to the usage of probability based node selection methodology, randomness is achieved in the system, thereby providing security to the packet being transferred.

REFERENCES

[1] Gopinath Ganapathy and S. Shakila, "Fuzzy Based Optimized Circuit Construction for Privacy Enhanced Onion Routing" . *European Journal of Scientific Research*. June 2014, Vol 123 Issue 2, ISSN 1450-216X/1450-202X, pp. 157-168

[2] Saaty, Thomas L, "Relative Measurement and its Generalization in Decision Making: Why Pairwise Comparisons are Central in Mathematics for the Measurement of Intangible Factors – The Analytic Hierarchy/Network Process". *Review of the Royal Academy of Exact, Physical and Natural Sciences, Series A: Mathematics (RACSAM)* 102 (2): 251–318. June 2008. Doi:10.1007/bf03191825. Retrieved 2008-12-22.

[3] Fishburn, P.C, "Additive Utilities with Incomplete Product Set: Applications to Priorities and Assignments". *Operations Research Society of America (ORSA)*, Baltimore, MD, U.S.A. 1967.

[4] Triantaphyllou, E., "Multi-Criteria Decision Making: A Comparative Study". *Dordrecht, the Netherlands: Kluwer Academic Publishers (now Springer)*. 2000. p. 320.ISBN 0-7923-6607-7.

[5] http://en.wikipedia.org/wiki/Cumulative_distribution_function

[6] Fred Glover, "Tabu Search - Part 1". *ORSA Journal on Computing 1 (2)*: 190–206.1989. doi:10.1287/ijoc.1.3.190.

[7] Fred Glover, "Tabu Search - Part 2". *ORSA Journal on Computing 2 (1)*: 4–32.1990. doi:10.1287/ijoc.2.1.4.

[8] Zhou, Peng, Xiapu Luo, and Rocky KC Chang. "Inference attacks against trust-based onion routing: Trust degree to the rescue." *Computers & Security* 39. 2013. 431-446.

[9] Catalano, Dario, et al. "Fully non-interactive onion routing with forward secrecy." *International journal of information security* 12.1 .2013. 33-47.

[10] Anderson, R. "Two remarks on public key cryptology". Invited Lecture, ACM-CCS'97. 1997.

[11] Canetti, R., Halevi, S., Katz, J. "A forward-secure public key encryption scheme. In: Advances in cryptology—EUROCRYPT". 2003. LNCS vol. 2656, pp. 255–271.

[12] Boneh, D., Boyen, X., Goh, E., "Hierarchical identity based encryption with constant size ciphertexts In: Advances in Cryptology—Eurocrypt. 2005. LNCS vol. 3494, 440–456.

[13] Backes, Michael, et al. "Provably secure and practical onion routing."*Computer Security Foundations Symposium (CSF), 2012 IEEE 25th*. IEEE, 2012.

[14] Zhou, Peng, et al. "SGor: Trust graph based onion routing." *Computer Networks*57.17. 2013. 3522-3544.

[15] Castillo-Pérez, Sergio, and Joaquin Garcia-Alfaro. "Onion routing circuit construction via latency graphs." *Computers & Security* 37. 2013.197-214.

[16] Panchenko, Andriy, and Johannes Renner. "Path selection metrics for performance-improved onion routing." *Applications and the Internet, 2009. SAINT'09. Ninth Annual International Symposium on*. IEEE, 2009.

[17] S. Shakila and Gopinath Ganapathy, "A Survey On Anonymity Based Solutions For Privacy Issues In Web Mining", International Journal of Computational Intelligence and Information Security, January 2014 Vol. 5, No. 1 ISSN: 1837-7823.

[18] S. Shakila and Gopinath Ganapathy, "Privacy for Interactive web browsing : A Study on Anonymous Communication protocols", International Journal of Advance Research in Computer Science and Management Studies, May 2014 Vol. 2 Issue 5, ISSN:2321-7782

AUTHORS PROFILE

S. Shakila is the Assistant Professor in the Department of Computer Science, Government Arts College, Tiruchirappalli, India. She obtained her Bachelors degree and Masters degree from Bharathidasan University, Tiruchirappalli, India in 1988 and 1990 respectively. She also did her Master's Degree in Engineering from Anna University, Chennai, India. Received Master of Philosophy in Computer Science from Bharathidasan University, Tiruchirppalli, India. Registered for Ph.D programme in Bharathidasan University, India. Her Research Interest includes Information Security, Distributed Computing and Grid Computing

Dr. Gopinath Ganapathy is the Professor & Head, Department of Computer Science and Engineering in Bharathidasan University, India. He obtained his under graduation and post-graduation from Bharathidhasan University, India in 1986 and 1988 respectively. He submitted his Ph.D in 1996 in Maduarai Kamaraj University, India. Received Young Scientist Fellow Award for the year 1994 and eventually did the research work at IIT Madras. He published around 40 research papers. He is a member of IEEE, ACM, CSI, and ISTE. He was a Consultant for a 8.5 years in the international firms in the USA and the UK, including IBM, Lucent Technologies (Bell Labs) and Toyota. His research interests include Security, Semantic Web, NLP, Ontology, and Text Mining.