# DYNAMIC FIREWALL CONFIGURATION FOR VIRTUAL NETWORKS:

# A COMPREHENSIVE AUTOMATION FRAMEWORK

**A. Lakshmipathi Rao[1], Palagati Anusha[2]**

*[1]Assistant Professor, Department of CSE, Guru Nanak Institute of Technology, Hyderabad*
*[2] Assistant Professor, Department of CSE, Guru Nanak Institute of Technology, Hyderabad*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** In recent years, segment routing has emerged as a revolutionary traffic engineering architecture. Segment routing, however, results in control overheads since more packet headers need to be included. When segment headers get excessively long, the overheads can significantly lower the forwarding efficiency for a big network. We suggest the intelligent routing scheme for traffic engineering (IRTE), which can provide load balancing with minimal control overheads, in order to meet the better of two goals. We first frame the problem as a mapping problem that maps various flows to important diversion sites in order to obtain optimal performance. Second, by reducing the problem to a k-dense subgraph problem, we demonstrate that it is nondeterministic polynomial (NP)-hard. We create improved ant colony optimisation (IACO), a popular ant colony optimisation technique for network optimisation problems, to address this issue. We also develop and examine the theoretical performance of the load balancing algorithm with diversion routing (LBA-DR). Ultimately, we test the IRTE in various real-world topologies, and the findings demonstrate that the IRTE works better than previous algorithms. For example, while testing on Bell Canada topology, the maximum bandwidth is 24.6% less than that of traditional algorithms.

Keywords: Segment Routing, Traffic Engineering, Control Overheads, Packet Headers, Forwarding Efficiency, Intelligent Routing Scheme.

## 1. INTRODUCTION:

Research interest in traffic engineering (TE) has always been high. IP routing protocols, routing optimisation issues, IP network overlays, etc. were the main topics of traditional TE. These experiments were mostly carried out in conventional IP networks. Researchers started concentrating on TE difficulties in the software defined network (SDN) with the introduction of traffic splitting and SDN protocol architecture. SDN can assist in achieving effective network management, which can resolve significant TE problems that are challenging to resolve in conventional networks. Still, there are several obstacles facing SDN, including as scalability problems that restrict its range of applications. Furthermore, segment routing (SR) offers benefits related to network architecture that can assist in resolving these issues in SDN. As a result, numerous academics started looking at the potential of fusing segment routing with SDN. In recent years, segment routing—a revolutionary network architecture—has made SDN even more controllable. In addition to becoming an implementation solution for particular TE issues in the SDN, SR can achieve compatibility with the SDN. Directional data transmission from the source node to the destination node is possible using SR. Nevertheless, in numerous circumstances, the control overhead in SR is likewise infinite, leading to inefficient data transmission. Meanwhile, the issue of an excessively big packet header in SR is disregarded by current solutions. As a result, a routing strategy that takes into account the SR's packet length limitation and achieves load balancing must be investigated. In TE, control overhead is a significant issue. The performance of network transmission can be optimised with the right overhead control systems. With the emergence of new network paradigms like as SDN, SR, and blockchain in recent years, control overhead optimisation has been applied in many contexts. But in the SDN and SR, control overhead optimisation is disregarded. An infinite control overhead results in an excessively long SR packet and reduces the effectiveness of data transfer. The control overhead issue in SR deteriorates as the routing length increases because an SR packet header grows longer. When researchers solved the problem of bandwidth load balancing in SR, they did not optimise network performance from the perspective of limited control overhead. Thus, further research into the TE technique in SR that combines reduced control overhead and bandwidth load balancing is warranted. In this work, we suggest using an SR setting. In an SR environment, the IRTE not only does bandwidth load balancing but also takes packet lengths into account for minimal control overhead. We suggest using the diversion routing technique to meet our goal of load balancing bandwidth. We install nodes with diversion routing capabilities and build the IRTE architecture using the SR controller. Every time interval, the controller gathers the network status and creates an information matrix. The controller computes the diversion routing paths for source hosts based on load balancing algorithms and network conditions after receiving their data transmission demand. For the IRTE, we create two brand-new overload optimisation and load balancing algorithms: improved ant colony optimisation (IACO) and load balancing algorithm with diversion routing (LBA-DR). After analysing this matrix, IACO offers a diversion routing path, which deviates from the conventional shortest routing method in most cases. On the one hand, this diversion routing method from these two algorithms lowers the network's maximum bandwidth usage while taking the bandwidth load balancing target into account. However, it prevents forwarding overload by managing the length of the SR packet header. Subsequently, the source transmits data to the destination and initiates data flow based on this diversion routing path. Lastly, the nodes on this routing send the SR co. an update on the network's current state. We juxtapose the created algorithms against conventional algorithms. We evaluate their performances across six common topologies with varying flow counts. We confirm that our simulation is a test-bed experiment by keeping track of each algorithm's control overhead. We evaluate the load balancing performances of these two algorithms and examine their trend changes. We also investigate how the number of nodes, degrees, and parameters of IACO relate to the algorithm's performance. The experimental findings demonstrate that the

IRTE achieves the bandwidth load balancing goal in various topologies. as utilising the LBA-DR or IACO, the maximum bandwidth used in most networks is decreased by 2.45% to 24.6% as compared to earlier algorithms. IACO achieves a performance advantage of 1.3% to 22.5% with increasing flow numbers, while LBA-DR achieves 2.1% to 17%. Furthermore, the load balancing gap between IACO and a traditional algorithm improves from 14.5% to 22.7% and from 9.34% to 29.3%, respectively, with changes made to ˛ and ˇ in IACO. In the meanwhile, one major limitation in every experiment is the length of the packet header.

## 2. LITERATURE SURVEY

Y. Xue et. al., explained that in public cloud storage services, data are outsourced to semi-trusted cloud servers which are outside of data owners' trusted domain. To prevent untrustworthy service providers from accessing data owners' sensitive data, outsourced data are often encrypted. In this scenario, conducting access control over these data becomes a challenging issue. Attribute-based encryption (ABE) has been proved to be a powerful cryptographic tool to express access policies over attributes, which can provide a fine-grained, flexible, and secure access control over outsourced data. However, the existing ABE-based access control schemes do not support users to gain access permission by collaboration. In this paper, we explore a special attribute-based access control scenario where multiple users having different attribute sets can collaborate to gain access permission if the data owner allows their collaboration in the access policy. Meanwhile, the collaboration that is not designated in the access policy should be regarded as a collusion and the access request will be denied. We propose an attribute-based controlled collaborative access control scheme through designating translation nodes in the access structure. Security analysis shows that our proposed scheme can guarantee data confidentiality and has many other critical security properties. Extensive performance analysis shows that our proposed scheme is efficient in terms of storage and computation overhead.

M. Ul Hassan et. al., discussed that Modern smart homes are being equipped with certain renewable energy resources that can produce their own electric energy. From time to time, these smart homes or micro grids are also capable of supplying energy to other houses, buildings, or energy grid in the time of available self- produced renewable energy. Therefore, researches have been carried out to develop optimal trading strategies, and many recent technologies are also being used in combination with micro grids. One such technology is block chain, which works over decentralized distributed ledger. In this paper, we develop a block chain based approach for micro grid energy auction. To make this auction more secure and private, we use differential privacy technique, which ensures that no adversary will be able to infer private information of any participant with confidence. Furthermore, to reduce computational complexity at every trading node, we use consortium block chain, in which selected nodes are given authority to add a new block in the block chain. Finally, we develop differentially private Energy Auction for block chain-based micro grid systems (DEAL). We compare DEAL with Vickrey-Clarke-Groves (VCG) auction scenario and experimental results demonstrates that DEAL outperforms VCG mechanism by maximizing sellers' revenue along with maintaining overall network benefit and social welfare.

G. M. Hastig and M. S. Sodhi discussed in their paper that seek to guide operations management (OM) research on the implementation of supply chain traceability systems by identifying business requirements and the factors critical to successful implementation. We first motivate the need for implementing traceability systems in two very different industries—cobalt mining and pharmaceuticals—and present business requirements and critical success factors for implementation. Next, we describe how we carried out thematic analysis of practitioner and scholarly articles on implementing block chain for supply chain traceability. Finally, we present our results pertaining to the needs of different stakeholders such as suppliers, consumers, and regulators. The business requirements for traceability systems are curbing illegal practices; improving sustainability performance; increasing operational efficiency; enhancing supply-chain coordination; and sensing market trends. Critical success factors for implementation are companies' capabilities; collaboration; technology maturity; supply chain practices; leadership; and governance of the traceability efforts. These findings provide a nascent measurement model for empirical work and a foundation for descriptive and normative research on block chain applications for supply chain traceability.

**Existing System:**

The characteristic values that suppliers gather during the production process are referred to as aviation supplier process quality data. These values are then utilised to derive the aviation product quality indicators. This data covers a broad range of subjects, such as return maintenance, production batches, usage, and manufacturing process planning. Aviation goods must have stable traceability storage, high-speed safe transmission, and fast and reliable quality characteristic data gathering in order to guarantee their overall performance and service life. However, the credibility sharing and security interoperability of quality data in the manufacturing process have always been a challenge in supplier quality management and control due to the involvement of aviation enterprises in a variety of product types, a large number of suppliers, and deep levels of the supply chain in the actual manufacturing process. For aircraft companies looking to raise their standard of supplier quality control, this has also been a barrier. Manufacturers and consumers alike are finding that they increasingly want a safe and dependable supply chain system. It is imperative to investigate a novel approach of profoundly integrating next-generation information technology with the aircraft manufacturing industry.

**Proposed System:**

An aircraft supplier manufacturing process quality data-sharing network based on blockchain is proposed in this article. The study first discusses the potential integration of block chain technology and manufacturing supply chain quality management. Second, based on quality state and island types of aviation suppliers, the architecture of the quality and data sharing platform of the production process of new aviation suppliers is described. Subsequently, a comprehensive approach is suggested to facilitate the real-time and systematic functioning of the sharing platform by implementing quality and data security sharing. On this basis, develop vital technologies including supplier assessment models, data storage security sharing, and manufacturing quality data block packaging models. In conclusion, the platform architecture and technology oversee shared application practices based on a particular aircraft industrial park, relying on data collection of supplier product production processes. By combining the data supply and request components, the application platform offers useful and astute methods for exchanging airline product quality data.

**Proposed System Advantage:**

- The orderly and real-time functioning of the sharing platform.
- Models for supplier assessment, security sharing, and data storage.
- Supplying airlines with sensible and useful sharing solutions.

**System Architecture**

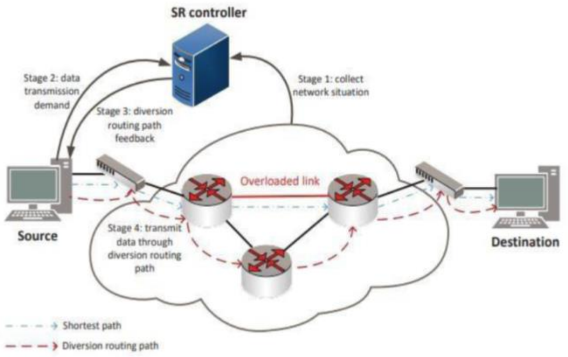

SR controller
Stage 2: data transmission demand
Stage 3: diversion routing path feedback
Stage 1: collect network situation
Overloaded link
Source
Destination
Stage 4: transmit data through diversion routing path
Shortest path
Diversion routing path

*Figure 1: System Architecture*

The data owner for this project must register all details before logging in. A document may be uploaded by the data owner. The data user may get a request from the data owner. A data user can utilise an uploaded document to search for a query. There is a download option for the file, which displays the encryption format. The cloud server receives requests from data users as well. A cloud server may have a login. The key will be accepted by it. The entire data set is visible to the cloud server as well. All user data is visible to the cloud server as well. The whole stored data set is visible to the cloud server. A key request from the user may be approved by the cloud server. After receiving the request, the data owner might provide the user with a secret key. After then, the user can download a file. The user receives a notice that they have a block if they enter the incorrect keys.

## 3. METHODOLOGY

**Modules:**

**User interface design:** We create the project's windows in this module. All users can securely log in using these windows. Users can only connect to the server by providing their login and password in order to establish a connection. The user can log in straight to the server if they have previously left; otherwise, they must register their information, including their email address, password, and username. In order to maintain the upload and download rates, the server will create an account for each user. The user ID will be set to name. Typically, logging in allows access to a certain page.

**SR Controller:** In this module users can sign up and log in is this one. Users have the opportunity to search files by name after logging in. Users of the data can also download a file that displays encrypted data. A trapdoor request can also be sent to the server by a data user. After the server approves the request, the user can obtain the owner's permissions and download the file in plain text.

**Segment Routing :** The data owner has to log in and register for this module. The files will be uploaded into the database by the data owner. Requests may be sent from the data owner to the data user.

**Control Overhead optimization: Login** to Cloud Server via this module. All data owners' details will be visible after logging in. The cloud server has access to every user's data. Every stored data file is visible to the cloud server. The user can request keys from the cloud server. The file metadata of an attacker is likewise visible to the cloud server.

## 4. IMPLEMENTATION:

**Intelligent routing scheme for traffic engineering (IRTE)**

In this study, we propose an intelligent TE (IRTE) routing system in an SR setting. In an SR environment, the IRTE not only does bandwidth load balancing but also takes packet lengths into account for minimal control overhead. We suggest using the diversion routing technique to meet our goal of load balancing bandwidth. We install nodes with diversion routing capabilities and build the IRTE architecture using the SR controller. The matrix of information for every time interval. The controller computes the diversion routing paths for source hosts based on load balancing algorithms and network conditions after receiving their data transmission demand. For the IRTE, we create two brand-new overload optimisation and load balancing algorithms: improved ant colony optimisation (IACO) and load balancing algorithm with diversion routing (LBA-DR). Using randomised rounding and linear programming, the LBA-DR creates the diversion routing path for every source node. After analysing this matrix, IACO offers a diversion routing path, which deviates from the conventional shortest routing method in most cases. On the one hand, this diversion routing method from these two algorithms lowers the network's maximum bandwidth usage while taking the bandwidth load balancing target into account.

## 5. EXPERIMENTAL RESULTS:

Home Page

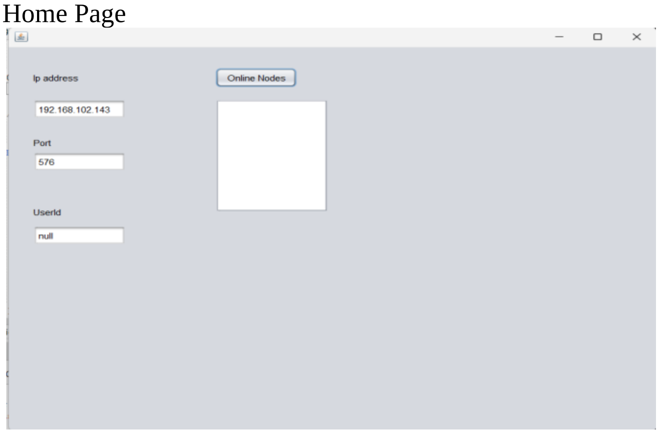

Ip address
192.168.102.143
Port
576
UserId
null

*Figure 2: Home Page*

The homepage Contains Ip address ,Port number , userid and online nodes.

**User Registration**

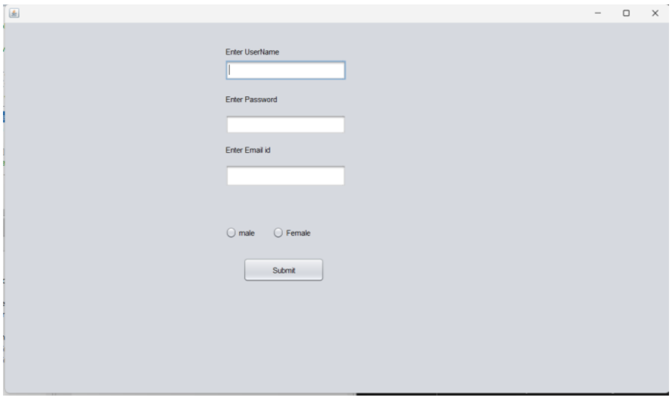


*Figure 3: User Registration*

Before login, user has to register with valid details.

**Login Page**

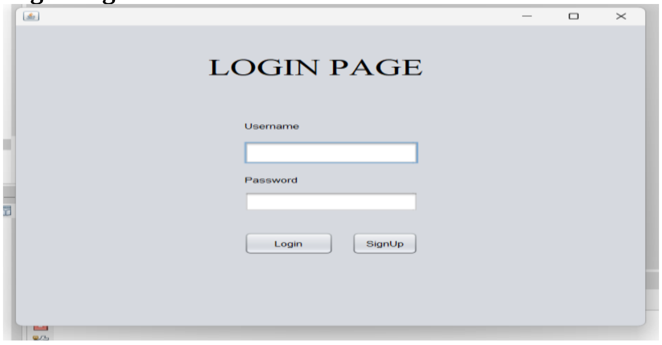
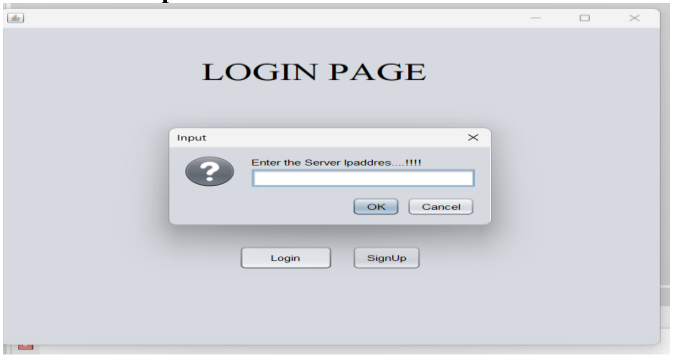


*Figure 4: Login Page*

In the login page user has to enter his/her username and password to login to the page

**Users Server Ip address**

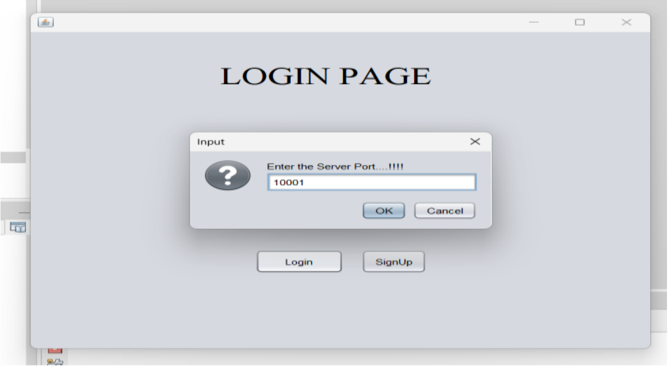


Enter the Server Ipaddres....!!!!

*Figure 5: Users Server Ip address*

While logging, user has to enter Server Ip address for security purpose

**User's Server Port no**



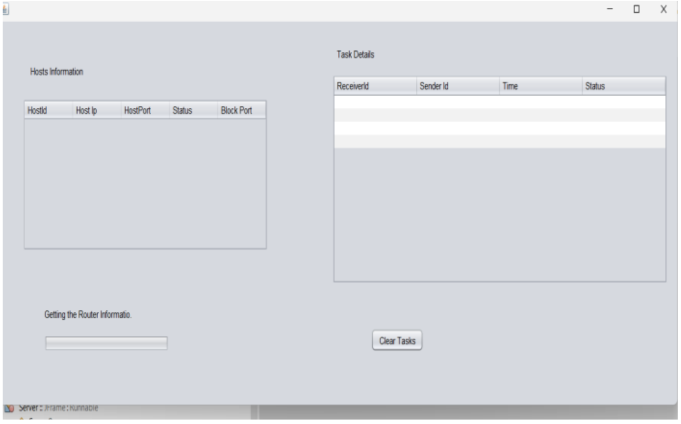
Enter the Server Port....!!!!

10001

*Figure 6: User's Server Port no*

Enter User Server Port number also before login.

**Server Status Before User Login**



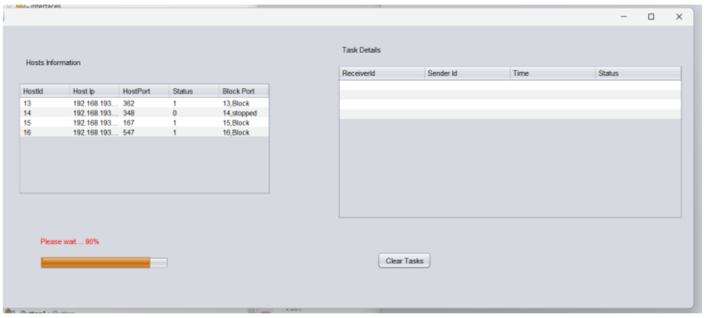
| HostId | Host Ip | HostPort | Status | Block Port |
|---|---|---|---|---|

| ReceiverId | Sender Id | Time | Status |
|---|---|---|---|

*Figure 7: Server Status Before User Login*

Before login the system will give server status.

**User's Dashboard**



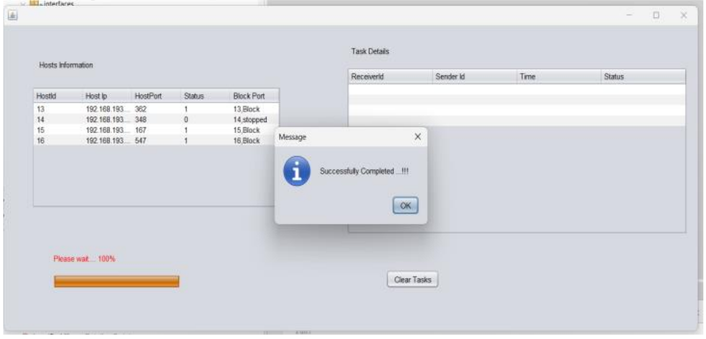
| HostId | Host Ip | HostPort | Status | Block Port |
|---|---|---|---|---|
| 13 | 192.168.193.... | 362 | 1 | 13,Block |
| 14 | 192.168.193.... | 348 | 0 | 14,stopped |
| 15 | 192.168.193.... | 167 | 1 | 15,Block |
| 16 | 192.168.193.... | 547 | 1 | 16,Block |

| ReceiverId | Sender Id | Time | Status |
|---|---|---|---|

Please wait.... 86%

*Figure 8: User's Dashboard*

**Blocking the User port:**



| HostId | Host Ip | HostPort | Status | Block Port |
|---|---|---|---|---|
| 13 | 192.168.193.... | 362 | 1 | 13,Block |
| 14 | 192.168.193.... | 348 | 0 | 14,stopped |
| 15 | 192.168.193.... | 167 | 1 | 15,Block |
| 16 | 192.168.193.... | 547 | 1 | 16,Block |

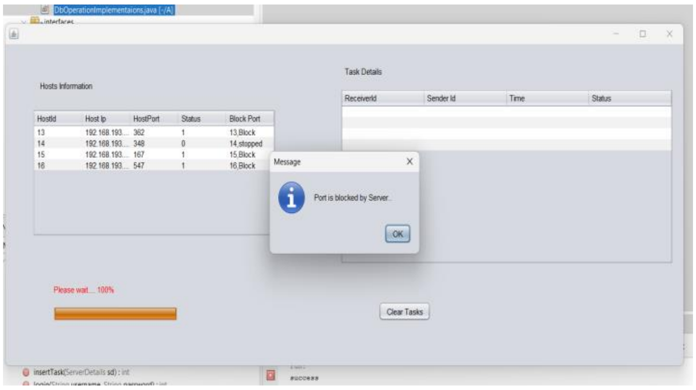
| ReceiverId | Sender Id | Time | Status |
|---|---|---|---|

Successfully Completed ....!!!

Please wait.... 100%

*Figure 9: Blocking the User port*

**User Port is Blocked**


DbOperationImplementaions.java [-/A]
interfaces
Hosts Information
Task Details
Message
OK
Clear Tasks
insertTask(ServerDetails sd) : int
success

| HostId | Host Ip | HostPort | Status | Block Port |
|---|---|---|---|---|
| 13 | 192.168.193.... | 362 | 1 | 13,Block |
| 14 | 192.168.193.... | 348 | 0 | 14,stopped |
| 15 | 192.168.193.... | 167 | 1 | 15,Block |
| 16 | 192.168.193.... | 547 | 1 | 16,Block |

| ReceiverId | Sender Id | Time | Status |
|---|---|---|---|

Port is blocked by Server..

Please wait.... 100%

*Figure 10: User Port is Blocked*

## 6. CONCLUSION:

In this study, we present the IRTE to provide load balancing in the SR context with low control overheads. In our IRTE system design, we present the role of the hosts and SR controller. We formulate the data flow transmission mapping problem and suggest the concept of diversion routing. We establish the NP-hardness of the diversion routing problem. Furthermore, in order to achieve minimum control overheads and bandwidth load balancing, we design LBA-DR and IACO algorithms. We assess the IRTE in several real-world topologies and contrast its performance with that of the conventional routing algorithm. The results demonstrate that the IRTE delivers low control overheads in SR with superior load balancing performance than the standard approach.

## 7. FUTURE ENHANCEMENT:

In order to update the network information matrix, the nodes on this routing route finally communicate the most recent network status to the SR controller. Moreover, security operations can be made more efficient and streamlined by automating the management and optimisation of firewall rules. In response to shifting network conditions, application needs, and security regulations, automated systems can dynamically modify firewall rules, eliminating the need for manual intervention and lowering the possibility of setup errors. Furthermore, implementing automatic firewalls in virtual networks can be made easier with the adoption of Software-Defined Networking (SDN) concepts. The deployment of security policies in virtualized environments can be made more flexible and scalable thanks to SDN's ability to provide centralised management and programmability of network resources.

## REFERENCES:

[1] Y. Xue, K. Xue, N. Gai, J. Hong, D. S. L. Wei and P. Hong, "An Attribute-Based Controlled Collaborative Access Control Scheme for Public Cloud Storage," in *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 11, pp. 2927-2942, Nov. 2019.

[2] M. U. Hassan, M. H. Rehmani and J. Chen, "DEAL: Differentially Private Auction for Blockchain-Based Microgrids Energy Trading," in *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 263-275, 1 March-April 2020.

[3] Hastig, Gabriella & Sodhi, Manmohan. (2019). Blockchain for Supply Chain Traceability: Business Requirements and Critical Success Factors. Production and Operations Management. 29. 10.1111/poms.13147.

[4] D. O. Awduche and B. Jabbari, Internet traffic engineering using multi-protocol label switching (MPLS), Comput. Netw., vol. 40, no. 1, pp. 111–129, 2002.

[5] N. Wang, K. H. Ho, G. Pavlou, and M. Howarth, An overview of routing optimization for internet traffic engineering, IEEE Commun. Surv. Tut., vol. 10, no. 1, pp. 36–56, 2008.

[6] Y. Wang, Z. Wang, and L. Zhang, Internet traffic engineering without full mesh overlaying, in Proc. IEEE INFOCOM 2001. Conf. Computer Communications. Twentieth Annual Joint Conf. IEEE Computer and Communications Society (Cat. No. 01CH37213), Anchorage, AK, USA, 2001, pp. 565–571.

[7] A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, A survey on the contributions of software-defined networking to traffic engineering, IEEE Commun. Surv. Tut., vol. 19, no. 2, pp. 918–953, 2017.

[8] M. Karakus and A. Durresi, A survey: Control plane scalability issues and approaches in software-defined networking (SDN), Comput. Netw., vol. 112, pp. 279–293, 2017.

[9] S. H. Yeganeh, A. Tootoonchian, and Y. Ganjali, On scalability of software-defined networking, IEEE Commun. Mag., vol. 51, no. 2, pp. 136–141, 2013.

[10] P. L. Ventre, M. M. Tajiki, S. Salsano, and C. Filsfils, SDN architecture and southbound APIs for IPv6 segment routing enabled wide area networks, IEEE Trans. Netw. Serv. Manag., vol. 15, no. 4, pp. 1378–1392, 2018.

[11] Z. N. Abdullah, I. Ahmad, and I. Hussain, Segment routing in software defined networks: A survey, IEEE Commun. Surv. Tut., vol. 21,no. 1, pp. 464–486, 2019.

[12] J. Zhang and C. Zhao, Q-SR: An extensible optimization framework for segment routing, Comput. Netw., vol. 200, p. 108517, 2021.

[13] X. Li and K. L. Yeung, Bandwidth-efficient network monitoring algorithms based on segment routing, Comput. Netw., vol. 147, pp. 236–245, 2018.

[14] X. Jia, Q. Li, Y. Jiang, Z. Guo, and J. Sun, A low overhead flow-holding algorithm in software-defined networks, Comput. Netw., vol. 124, pp. 170–180, 2017.

[15] R. Banerjee and S. Das Bit, Low-overhead video compression combining partial discrete cosine transform and compressed sensing in WMSNs, Wirel. Netw., vol. 25, no. 8, pp. 5113– 5135, 2019.

[16] A. Cianfrani, M. Listanti, and M. Polverini, Incremental deployment of segment routing into an ISP network: A traffic engineering perspective, IEEE/ACM Trans. Netw., vol. 25, no. 5, pp. 3146–3160, 2017.

[17] X. Xiao, A. Hannan, B. Bailey, and L. M. Ni, Traffic engineering with MPLS in the internet, IEEE Netw., vol. 14, no. 2, pp. 28–33, 2000.

[18] B. Fortz, J. Rexford, and M. Thorup, Traffic engineering with traditional IP routing protocols, IEEE Commun. Mag., vol. 40, no. 10, pp. 118–124, 2002.

[19] A. Elwalid, C. Jin, S. Low, and I. Widjaja, Mate: MPLS adaptive traffic engineering, in Proc. IEEE INFOCOM 2001, Conf. Computer Communications, Twentieth Annual Joint Conf. IEEE Computer and Communications Societies, Anchorage, AK, USA, 2001, pp. 1300–1309.

[20] A. Feldmann and J. Rexford, IP network configuration for intradomain traffic engineering, IEEE Netw., vol. 15, no. 5, pp. 46–57, 2001.

## BIOGRAPHIES

Mr. A. LAKSHMIPATHI RAO is working as Asst. Professor at Guru Nanak Institute of Technology, Hyderabad. He completed M.Tech CSE with Distinction from JNTUH, Kukatpally in 2010. He has 14+ years of teaching experience. He has published 03 international journals and attended 25 FDPs, 4 National level workshops and 1 International workshop and published text book of Data Mining and Data Warehusing and has 2 patents. He was a Member of IAENG, Membership Number: 162690. He has Qualified GATE with 88.68 percentile

Mrs. Palagati Anusha is working as Assistant Professor in the Department of Computer Science and Engineering at Guru Nanak Institute of Technology, Hyderabad with a diverse educational background. She completed her B.Tech IT from JB Women's Engineering College, Tirupati in 2012, M.Tech CSE with Distinction from Geethanjali Institute of Technology, Nellore in 2015, and is currently pursuing a Ph.D. at Anna University, Chennai. She has 8 years of experience in teaching field. She has made significant contributions to her field with 11 articles published in reputed journals, 7 patents and she has attended 5 conferences at national and international levels.