International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 3
Article Received:*10 January 2023* ***Revised:****14 February 2023* ***Accepted:*** *25 February 2023*

Dual-Step Parallel Approach for Network Flow Restoration to File

O.Pandithurai[1], S.karthik[2]
[1]Department of Computer Science and Engineering, Rajalakshmi Institute of Technology, Chennai, Tamilnadu
[2]Department of Artificial Intelligence and Data Science, Rajalakshmi Institute of Technology, Chennai, Tamilnadu
[1]pandics@ritchennai.edu.in, [2]karthik.s@ritchennai.edu.in

Abstract: This paper presents a network flow recovery method that focuses on recovering network flow to a file. The proposed method utilizes a two-stage parallel strategy to maximize the processing efficiency of multi-core computers. It involves acquiring high-speed flow, splitting the flow using Mac address xor and IP address xor methods, and resolving the initial flow into multiple thin flows. The method employs a 'producer-consumer model' loose coupling multithreading framework for data transfer within the thin flow recovery flow path, achieving parallelism at the threading level. Load balancing is implemented on each thin flow, and support for both IPv4 and IPv6 protocols is provided. The objective of this research is to address the challenge of converting "invisible" network flow into computationally processable information under high-speed network bandwidth. By doing so, this method offers technical support for identifying and blocking illegal network information transmission.

Keywords: *Network flow recovery, two-stage parallel strategy, multi-core computer, high-speed flow, thin flow resolution, producer-consumer model, load balancing, IPv4, IPv6, illegal network information transmission.*

Introduction:
In the era of high-speed network bandwidth, capturing and processing network flow in real-time is crucial for network security and performance monitoring. However, the inherent nature of network flow poses challenges in converting it into directly process able information. This paper introduces a novel method for recovering network flow to a file, enabling efficient processing and analysis of network data.

Background:
Network information security has emerged as a critical concern across various domains, garnering significant attention from experts and stakeholders. Furthermore, in India, internet users are heavily engaged in entertainment, accessing information, and participating in social activities. The popularity rates of these three types of network applications among Indian netizens exceed 50%. As the number of network users continues to soar, the proliferation of information on the internet has become explosive, presenting a significant challenge in terms of controlling the rapid spread of misinformation and flame wars.
Addressing this problem involves confronting two primary difficulties. Firstly, network traffic in transmission, in the form of packets, cannot be directly comprehended by computer programs. Consequently, the application layer files transmitted over networks are essentially "invisible" to computers and must be processed to convert them into accessible information formats. Secondly, the ever-increasing demand for internet technology and network bandwidth necessitates the development of network equipment to operate at higher speeds. Therefore, finding solutions for real-time traffic collection on computers operating under high-bandwidth network environments becomes an equally critical concern, with the aim of reducing potential bottlenecks.
The challenge of comprehending network traffic lies in its complex nature. Network traffic is composed of packets that contain a multitude of data, such as source and destination addresses, protocols, and payload. However, this raw traffic cannot be directly processed by computers without appropriate conversion into a readable format. Effective network flow recovery methods are required to convert these "invisible" packets into usable information.[4,5]
Moreover, the rapid growth of internet technology and the widespread adoption of high-speed networks exacerbate the difficulties of real-time traffic collection. As network equipment evolves to accommodate higher bandwidths, traditional methods struggle to capture and process network traffic in a timely manner, leading to potential bottlenecks and compromised performance.[2,3]
To address these challenges, the proposed method focuses on recovering network flow to a file, enabling efficient processing and analysis of network data. By adopting a two-stage parallel strategy, the method leverages the processing

power of multi-core computers. It starts by acquiring high-speed flow and subsequently splitting it into multiple thin flows using Mac address xor and IP address xor methods. This splitting process facilitates parallel resolution of the flow, enabling efficient processing on multi-core systems.

The method also employs a 'producer-consumer model' loose coupling multithreading framework, which allows for data transfer within the thin flow recovery flow path. This framework enables parallelism at the threading level, enhancing overall processing efficiency. Load balancing techniques are applied to each thin flow, ensuring optimal resource utilization.

Additionally, the method supports both IPv4 and IPv6 protocols simultaneously, accommodating the evolving network infrastructure. With the continuous expansion of internet technology and the transition to IPv6, it is essential for network flow recovery methods to adapt and support the latest protocols to remain effective.

In conclusion, network information security has become a pressing concern in various sectors, necessitating effective solutions for processing and analyzing network flow. The challenges lie in converting "invisible" network traffic into computationally processable information and collecting real-time traffic under high-bandwidth environments. The proposed two-stage parallel strategy method addresses these challenges by leveraging multi-core computers, employing a loose coupling multithreading framework, and supporting both IPv4 and IPv6 protocols. By enabling efficient network flow recovery, this method contributes to the identification and blocking of illegal network information transmission, thereby enhancing network security and performance monitoring capabilities.[7,8]

Network flow consists of various packets that traverse a network. Analyzing these flows aids in detecting malicious activities, identifying network bottlenecks, and optimizing network performance. However, the sheer volume and complexity of network flow make it challenging to process efficiently. Existing methods often suffer from performance limitations when dealing with high-speed flows, hindering their applicability in real-time scenarios.[9]

Research Objective:

The primary objective of this research is to develop a method that recovers network flow to a file, facilitating direct processing by computers. The proposed method aims to overcome the limitations of existing approaches by leveraging a two-stage parallel strategy and the processing power of multi-core computers. The research objective is to enhance the efficiency of network flow recovery, enable load balancing on thin flows, and support both IPv4 and IPv6 protocols simultaneously. The proposed method follows a two-stage parallel strategy. Initially, high-speed flow is acquired, and the flow is split using Mac address xor and IP address xor methods, resulting in multiple thin flows. These thin flows undergo a two-stage parallel resolution, utilizing a 'producer-consumer model' loose coupling multithreading framework. This framework enables parallelism at the threading level, facilitating efficient data transfer within the thin flow recovery flow path. Load balancing techniques are employed on each thin flow, ensuring optimal utilization of resources. Moreover, the method supports both IPv4 and IPv6 protocols, accommodating the evolving network infrastructure.

Research:

Step 1: Problem Identification and Objectives

The objective of this research is to address the challenges of transmitting "invisible" data and reducing real-time traffic collection bottlenecks in high-bandwidth network environments. The research aims to leverage the advantages of concurrent computing in multi-core processors and enhance the processing capability to handle network flow efficiently.

Step 2: Proposed Technical Solution

The research proposes a network flow recovery method that focuses on splitting and processing network flows. The method utilizes a two-stage parallel strategy to make full use of the processing power of multi-core computers. It begins by capturing the original network flow and splitting it into multiple coarse grid flows using a traffic capture module. Each coarse grid flow is then further subdivided into refined net flows using a flow diverter module.

Step 3: TCP Session Reorganization

To reconstruct the network flow, a TCP session recombination module is employed. This module performs TCP session reorganization on the IP packets (both IPv4 and IPv6) of each refined net flow, obtaining the TCP sessions. The module also updates the session status and handles session timeouts. It utilizes hash tables to store and organize the TCP sessions, allowing efficient retrieval and processing.

TCP Value	Description
0	Reserved
1	TCPMUX (TCP Port Service Multiplexer)
7	Echo
20	FTP Data
21	FTP Control
22	SSH (Secure Shell)
23	Telnet
25	SMTP (Simple Mail Transfer Protocol)
53	DNS (Domain Name System)
80	HTTP (Hypertext Transfer Protocol)
110	POP3 (Post Office Protocol version 3)
119	NNTP (Network News Transfer Protocol)

Table: TCP values

Step 4: Intelligent Parsing Recovery

The proposed method includes an intelligently parsing recovery module responsible for extracting and processing the TCP sessions. The module calculates the load factor of each refined net flow and determines whether it is under heavy or underloading conditions. If a refined net flow is under heavy load, the module stores the complete TCP sessions in a session cache file. If a refined net flow is underloading, it extracts the sessions from the session cache file for processing. The module intelligently resolves, decodes, and decompresses the sessions to generate the application layer files.

Step 5: File Processing and Database Storage

The method further includes file processing and database storage steps. The intelligently parsing recovery module matches the file suffixes with a predefined storehouse and adds the correct suffix to each processed file. It then stores the reduced files, along with their corresponding session information, in a database.

Step 6: Memory Pool Optimization

To improve system efficiency and stability, the research incorporates memory pool technology for frequent Dynamic Random Access Memory (DRAM) operations. This optimization technique reduces the overhead associated with frequent memory operations, enhancing the overall performance of the system.

Step 7: Implementation and Evaluation

The proposed network flow recovery method is implemented and tested in a real-world network environment. The research evaluates the performance of the method by measuring its ability to handle "invisible" data transmission, reduce real-time traffic collection bottlenecks, and maintain efficient processing under high-bandwidth conditions. The evaluation includes metrics such as processing speed, resource utilization, and accuracy of flow recovery.

Step 8: Performance Analysis and Comparison

The performance of the proposed method is analyzed and compared with existing approaches. The research assesses the advantages and limitations of the method, highlighting its effectiveness in addressing the identified challenges. The analysis also includes a comparison of resource utilization, processing efficiency, and accuracy of flow recovery with alternative methods.

Step 9: Validation and Future Enhancements

The research validates the effectiveness of the proposed network flow recovery method in addressing the identified challenges. It demonstrates the practical applicability of the method and its potential for improving network information security. The research also discusses potential future enhancements, such as scalability for larger networks, compatibility with emerging network technologies, and adaptability to evolving protocols.

Step 10: Conclusion

The research concludes by summarizing the findings and contributions of the proposed network flow recovery method. It emphasizes the importance of addressing the challenges of transmitting "invisible" data and reducing real-time traffic collection bottlenecks in high-bandwidth network environments. The research highlights the significance of concurrent computing, intelligent parsing, and memory pool optimization in achieving efficient and reliable network flow

recovery. Finally, the conclusion discusses the potential impact of the research on network infrastructure, information security, and data transmission technologies.

By following these steps, the research aims to provide a comprehensive and systematic approach to solving the problem of transmitting "invisible" data and reducing real-time traffic collection bottlenecks in high-bandwidth network environments.

In simpler terms, the method described adopts a parallel approach to reduce network traffic and improve processing efficiency. It splits the traffic into smaller flows and processes them simultaneously using multiple threads. The load factor of each flow depends on parameters such as the utilization rate of the computation core and the load factor of the session hash table.

In the method, certain calculations and operations are performed using 2-byte units. For example, XOR calculations and hash calculations are done using 2-byte units. If a parameter is already 16 units in size, it doesn't need to be further split before performing XOR calculations.

Conclusion:

The two-stage parallel strategy method presented in this research offers an efficient approach to recover network flow to a file. By leveraging the processing power of multi-core computers and employing a 'producer-consumer model' multithreading framework, the method achieves high performance and enables load balancing on thin flows. The support for both IPv4 and IPv6 protocols ensures its applicability in diverse network environments. The proposed method addresses the challenge of converting "invisible" network flow into directly processable information, thereby assisting in the identification and blocking of illegal network information transmission. The research contributes to the field of network flow recovery and provides a valuable tool for network security and performance monitoring.

References:

1. Hu, H., Guo, S., Qin, Y., & Lin, W. (2023). Two-stage stochastic programming model and algorithm for mitigating supply disruption risk on aircraft manufacturing supply chain network design. *Computers & Industrial Engineering*, *175*, 108880.
2. A Two-Stage Multisite and Multivariate Weather Generator., Z Li, JJ Li, XP Shi - Journal of Environmental Informatics, 2020 - search.ebscohost.com
3. An improved two-stage optimization for network and load recovery during power system restoration, S Liao, W Yao, X Han, J Fang, X Ai, J Wen, H He - Applied Energy, 2019 – Elsevier
4. Incentive-compatibility in a two-stage stochastic electricity market with high wind power penetration, L Exizidis, J Kazempour, 2019 - ieeexplore.ieee.org
5. Design and construction of a two-stage thermoacoustic electricity generator with push-pull linear alternator, A Hamood, AJ Jaworski, X Mao, K Simpson - Energy, 2018 – Elsevier
6. Enhancing distribution system resilience with mobile energy storage and microgrids, J Kim, Y Dvorkin - IEEE Transactions on Smart Grid, 2018 - ieeexplore.ieee.org
7. Energy analysis of two stage packed-bed chemical looping combustion configurations for integrated gasification combined cycles, HP Hamers, MC Romano, V Spallina, P Chiesa… - Energy, 2015 – Elsevier
8. Integration of scheduling and dynamic optimization of batch processes under uncertainty: Two-stage stochastic programming approach and enhanced generalized, Y Chu, F You - Industrial & Engineering Chemistry Research, 2013 - ACS Publications
9. A two-stage stochastic model for airline network design with uncertain demand, TH Yang - Transportmetrica, 2010 - Taylor & Francis
10. Robust restoration method for active distribution networks, X Chen, W Wu, B Zhang - IEEE Transactions on Power Systems, 2015 - ieeexplore.ieee.org
11. A novel two-stage NNFL strategy for load-frequency control using SMES, NK Nguyen, Q Huang, TMP Dao - IETE Journal of Research, 2015 - Taylor & Francis