Mathematical Biosciences and Engineering

MBE, 19(2): 1775–1785.
DOI: 10.3934/mbe.2022083
Received: 28 September 2021
Accepted: 17 November 2021
Published: 16 December 2021

http://www.aimspress.com/journal/MBE

Research article

Artificial intelligence-based network traffic analysis and automatic optimization technology

Jiyuan Ren, Yunhou Zhang, Zhe Wang* and Yang Song

Northeast Branch of State Gird Corporation of China, #1, Yingpan North Street, Shenyang, Liaoning, MO 110180, China

* **Correspondence:** Email: wangzzhe936@126.com; Tel: +8613898806158; Fax: +86024-23125555.

Abstract: Network operation and maintenance (O & M) activities of data centers focus mainly on checking the operating states of devices. O & M engineers determine how services are running and the bearing capacity of a data center by checking the operating states of devices. However, this method cannot reflect the real transmission status of business data; therefore, engineers cannot fully comprehensively perceive the overall running conditions of businesses. In this paper, ERSPAN (Encapsulated Remote Switch Port Analyzer) technology is applied to deliver stream matching rules in the forwarding path of TCP packets and mirror the TCP packets into the network O & M AI collector, which is used to conduct an in-depth analysis on the TCP packets, collect traffic statistics, recapture the forwarding path, carry out delayed computing, and identify applications. This enables O & M engineers to comprehensively perceive the service bearing status in a data center, and form a tightly coupled correlation model between networks and services through end-to-end visualized modeling, providing comprehensive technical support for data center optimization and early warning of network risks.

Keywords: ERSPAN technology; artificial intelligence; traffic analysis; network strategy; automatic optimization

1. Convergence between ERSPAN and TCP

Encapsulated Remote Switch Port Analyzer (ERSPAN) encapsulates mirrored packets into GRE packets with protocol number 0x88BE and sends them throughout the layer 3-routed network to a remote monitoring device [1]. In an Internet environment, when the application layer protocols are

used to exchange data between applications running on different hosts, a reliable tunnel-like connection is required to ensure the soundness of data transfer. This requires the Transmission Control Protocol (TCP), a connection-oriented, reliable, and byte-stream-based communication protocol implementing on the transport layer, which allows data streams to be sent from one location to another. TCP initiates a three-way handshake to establish a connection between a TCP client and a TCP server. Once the connection is established, the two sides are able to transfer data [2,3]. The specific process is described as follows:

The Client sends a SYN packet (SEQ = x) to the Server and enters the SYN_SEND state.

The Server receives the SYN packet and sends back a SYN-ACK (SEQ = y, ACK = x + 1) packet and enters the SYN_RECV state.

The Client receives the SYN-ACK packet from the Server and sends back an ACK (ACK = y + 1) packet and enters the Established state.

A four-way handshake is required for terminating a TCP connection. The specific process is as follows:

After the Client completes the data sending task, it sends a FIN (SEQ = x) packet to cl

The Server receives this FIN packet and sends back a FIN-ACK (SEQ = y, ACK = x + 1).

It also sends an end-of-life flag to the application at the Server side. After some time, the application that receives the flag sends a CLOSE call to close its socket, then the TCP sends a FIN of its own.

The Client sends back a FIN-ACK (ACK = y + 1) to confirm the termination of the connection.

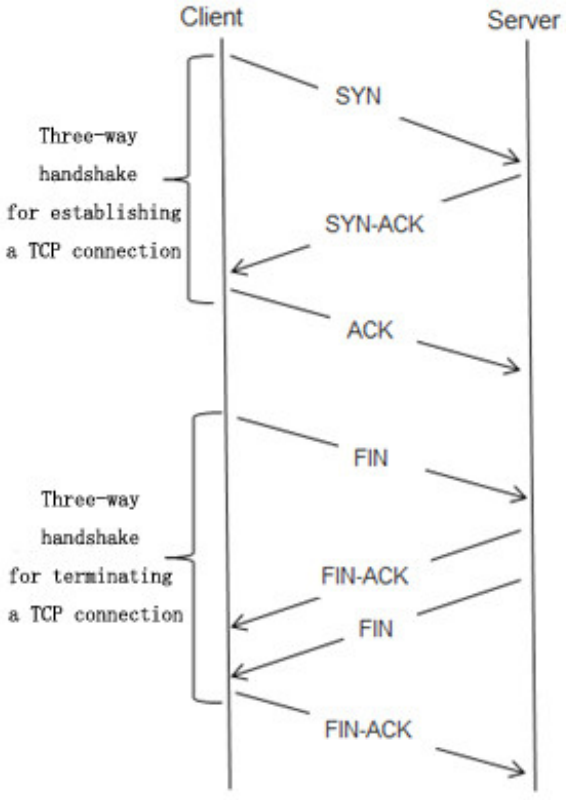

Figure 1. Packet interaction in TCP connection establishment and termination.

The source server initiates a request to establish a TCP connection with the target server. The TCP packet is passed through three routers (Laef1->Spine->Leaf2) before reaching the target server. The network path for forwarding TCP packets is denoted by the green dotted line in Figure 2. The ingress interface of each router along the path is configured as an ERSPAN source interface. A mirrored copy of the GRE-encapsulated TCP packet is transmitted by each router to the remote network AI collector, as indicated by the yellow dotted lines in Figure 2.

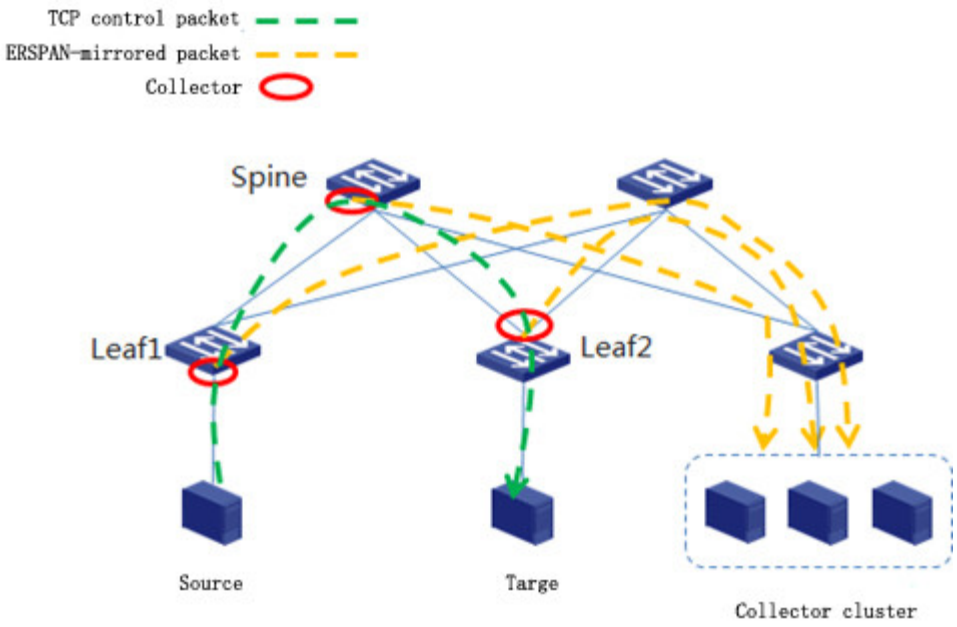

Figure 2. TCP control packet and ERSPAN mirrored packet convergence.

2. Session traffic statistics analysis

A network O & M AI analyzer is built to collect statistics on the size of a data stream in a TCP session based on the sequence number in the TCP headers of the SYN and FIN packets that starts and ends the whole session, respectively [4].

Traffic from the sender: sequence number in the FIN-ACK packet that is sent from the sender sequence number in the SYN packet that is sent from the sender

Traffic from the receiver: sequence number in the FIN-ACK packet that is sent from the receiver sequence number in the SYN-ACK packet that is sent from the receiver

The sequence number in the TCP header is a 32-bit field. As a TCP session is established, the value of the sequence number may count in reverse. If the backward counting only occurs once during the establishment of the TCP session, the algorithm for collecting the statistics can identify this during calculation and automatically correct the value (the minuend is smaller than the subtrahend). If the value of the sequence number count in reverse multiple times during the TCP session, the algorithm fails to identify, resulting in a large error in traffic statistics (the error is a multiple of 0xFFFFFFFF).

Restore the forwarding path of a TCP packet: Each time the TCP packet arrives at a layer 3 network device (a hop), the value of the TTL field in the IP packet header is reduced by one. Based on

this principle, the algorithm of the network O & M AI analyzer first matches all collected TCP packets based on the internal packet content and determines that the three TCP packets collected from three routers belong to the same TCP session. Then the analyzer sorts the packets in descending order based on the internal and external TTL values, followed by performing calculations based on the defined matching and identification rules to restore the forwarding path of the original TCP packet [5,6]. The restored forwarding path is Leaf1->Spine1->Leaf2, as shown in Figure 2.

Calculate the TCP packet latency: Packets in an ERSPAN-mirrored data stream cannot contain timestamp information; therefore, the time when a TCP data stream reaches each device is actually recorded by the timestamp inserted by the AI collector after packets in the data stream are mirrored to the collector. When the AI analyzer is performing the path restoration calculation, it calculates the TCP packet latency for each hop based on the timestamp inserted by the AI collector.

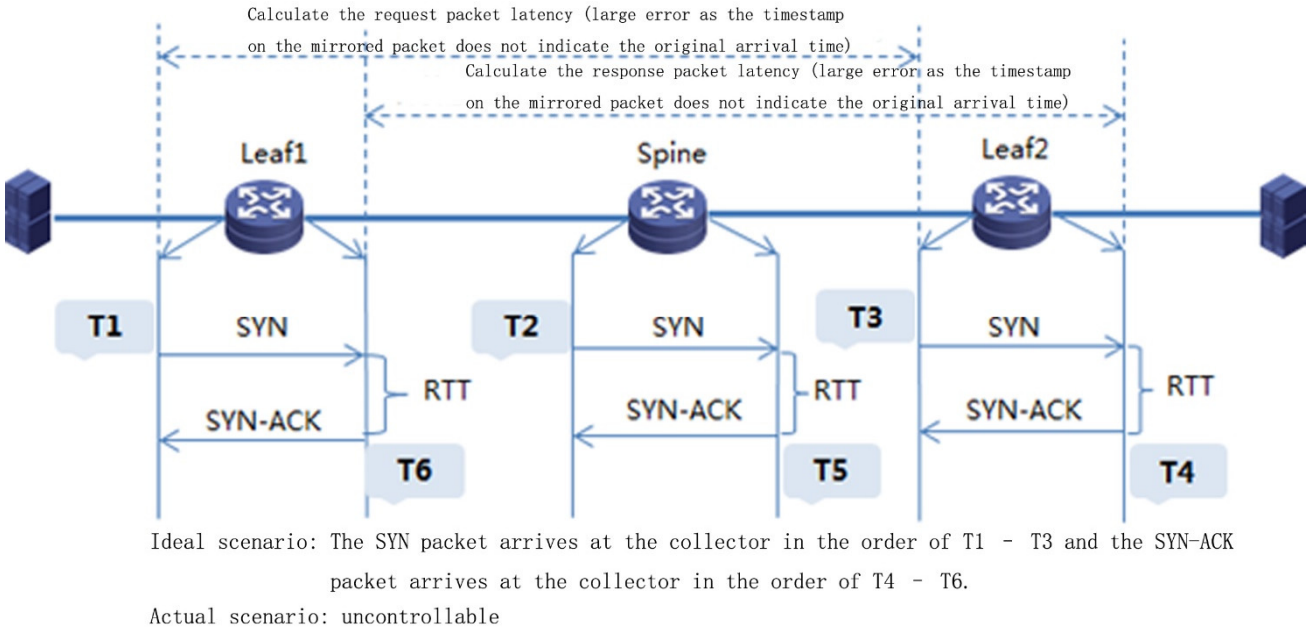

Figure 3. ERSPAN-mirrored data stream configuration.

As shown in Figure 3, to collect TCP SYN packets, an ERSPAN-mirrored stream of data packets is configured for routers. According to the timestamp inserted by the collector when it receives the TCP SYN packets mirrored from the three routers, the latency for each hop is T2–T1 and T3–T2. However, in the actual network environment, as transmission paths and collectors for handling data streams vary and delays exist in local caching of routers along the path, the time when the three mirrored packets arrive at the collector and the order in which the original packets reach each router are different. As a result, the timestamp inserted by the collector may be out of order, which further leads to the error difference between the latency obtained by calculation and the actual latency. For example, T2 may reach the collector earlier than T1. In this case, the timestamp value of T2 may be smaller than that of T1, and the calculated latency for a packet to transmit from Leaf1 to Spine is a quite larger value. (Similar to backward counting)

To prevent this issue, an ERSPAN-mirrored data stream should be configured for routers in the direction from the receiver to collect SYN-ACK packets. The network O & M AI analyzer calculates

the difference between the time a device receives SYN-ACK and the time it receives a SYN (RTT = T6–T1). If the RTT is greater than or equal to 0, the larger value obtained before is abnormal and should be treated as an invalid value. If the RTT is smaller than 0, then the larger value obtained before can be used.

TCP application identification: Based on the five-tuple information in internal TCP packets, along with the host number and port of the application layer, a specific application can be accurately identified. The application identification algorithm of the network O & M AI analyzer requires the user to first enter the IP and the name of the application to be identified. When processing TCP data streams, the analyzer automatically matches the user's input information with the five-tuple information of internal TCP packets and the host number and port of the application layer, so as to accurately identify which application has sent the TCP data stream. The analyzer also uses artificial intelligence algorithms for data analysis of the collected network traffic to proactively determine the corresponding relations between the five-tuple of packets, application layer information, and the application to be identified, thus forming an offline feature database [7]. While identifying the application that has sent the TCP packets, the database is automatically checked, which effectively avoids identification errors caused by incorrect user input.

Table 1. TCP exceptions that can be detected by the analyzer.

Exception Type	Description
connection exception	If the server returns no response within a set time after the client sends TCP packets (SYN, SYN-ACK, or FIN-ACK), the TCP retransmission mechanism will be triggered to retransmit the TCP packets.
	An Retransmission Timeout (RTO) occurs when SYN or SYN-ACK packets are sent; or the server responds directly by sending an RST packet on the same connection after the client sends an SYN packet. When an RTO occurs, the sender will continue to wait for two minutes. If FIN and RST packets are reported within two minutes, the connection is successfully established. If no other packets are received after two minutes, the connection fails to be established because the sender does not receive any SYN-ACK packets.
TCP session exception	RST flag is set to 1.
	The value of the internal TTL is less than 3.
	SYN and FIN flags are set at the same time. SYN and RST flags are set at the same time. FIN, PSH, and URG flags are set at the same time. SYN and PSH flags are set at the same time. The FIN flag is set while the ACK flag is not.

Int (in band network telemetry) is a network monitoring technology that collects data from devices. The equipment actively sends the collected data to the collector, provides real-time and high-speed data acquisition function, and achieves the purpose of monitoring the performance of network equipment and network operation.

Through int technology, the input and output port and queue information of each device on the message forwarding path, the timestamp information of the input and output devices, the congestion

information of the queue, etc. can be monitored; At the last hop of path detection, the monitored data is encapsulated with UDP header and IP header and forwarded to the collector. Finally, through the network management software deployed on the collector, the monitoring data are analyzed and useful information is extracted.

Through int technology, the accurate network forwarding path and delay of application flow can be presented, and the network health of applications in the network can be directly located.

Int networking is as follows.

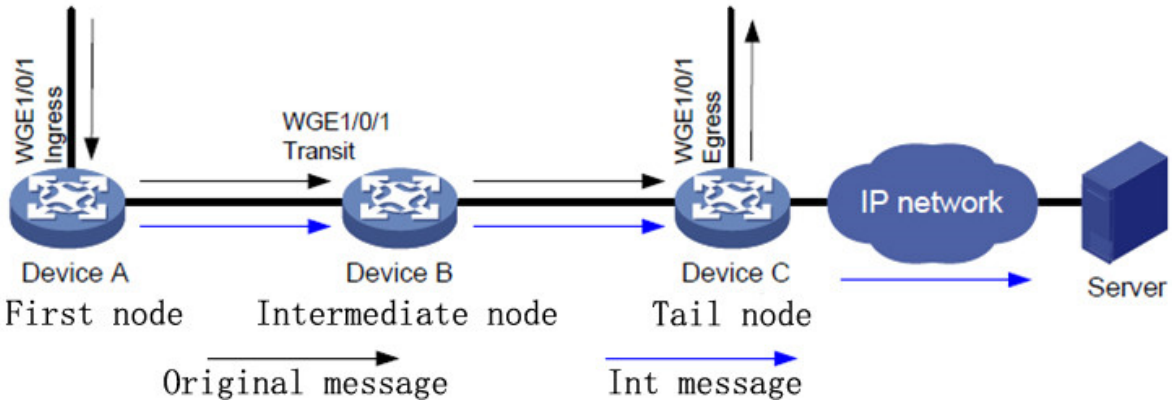

Figure 4. Int networking diagram.

In int network, the first device is the first node and the last hop device is the tail node. All devices except the first node and the tail node are intermediate nodes. Each int device plays different roles in the domain. The roles and their functions are as follows:

First node

The first node applies the QoS policy on the traffic inlet port to mirror the message conforming to the rules received on the interface to the int processor in the equipment. The processor adds int mark and timestamp to the message, generates int message and sends it to the intermediate node

Intermediate node

After receiving the message, the incoming port of the intermediate node automatically identifies the int message according to the int mark, sends it to the int processor in the equipment, adds a timestamp, and then sends it to the tail node.

Tail node

After receiving the message, the incoming port of the tail node automatically identifies the int message according to the int mark, and sends it to the int processor in the equipment to add a timestamp, and then encapsulates the int message according to the specified IP address, port number and VLAN number and sends it to the collector.

The following figure shows the schematic diagram of int network flow collection and int message packaging and forwarding (the red solid line is the original forwarding flow and the green dotted line is the int collection flow).

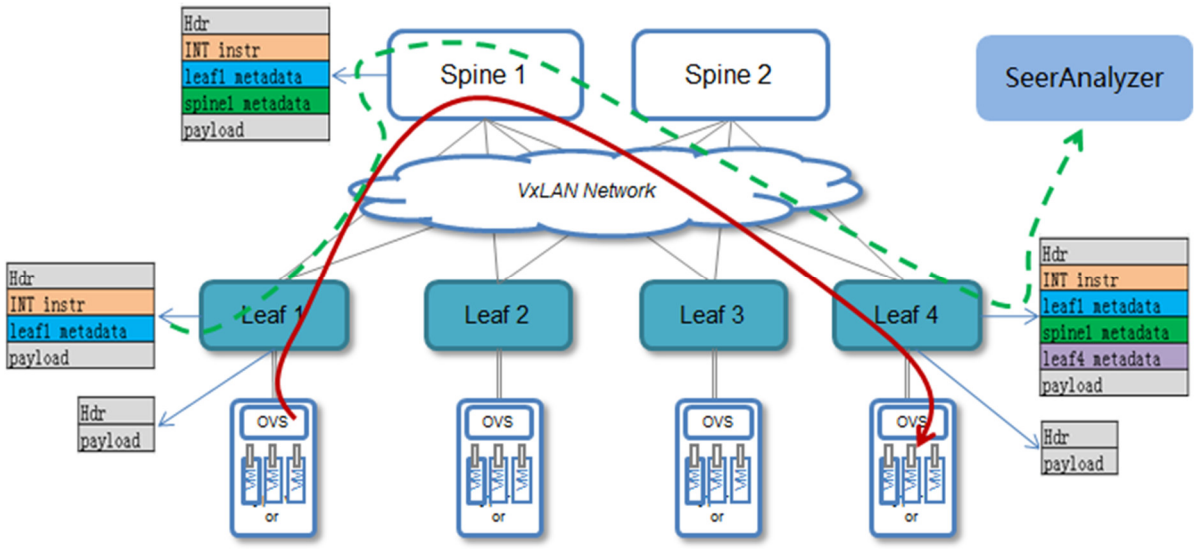

Figure 5. Int message encapsulation and forwarding diagram.

3. Network health monitoring

The health of network devices is a comprehensive measurement metric. To build a network health AI algorithm, we need to conduct a comprehensive correlative analysis from multiple dimensions according to the weight proportion of the network, user, and application, thereby getting a health score that can accurately reflect the operating states of the current network, user, and application [8]. When the health score is high, the operating state is relatively good. Otherwise, they are being operated in a subhealth state or even an abnormal state.

Network health: A 360-degree monitoring on networking devices is carried out from three dimensions: system plane, data plane and control plane. And the metrics are measured comprehensively according to certain weights to get a networking device health degree. The changing trend of the health degree represents the operating status of the device.

User health: A 360-degree monitoring on users access to the network is carried out from the dimensions of wireless service, authentication service, networking service and network management service. A comprehensive user health index is thus obtained. Wireless users and wired users are evaluated in the same way. The trend of the user health degree can show the login quality and connection quality of users access to the network resource.

Application health: A 360-degree monitoring on the application traffic and traffic quality is carried out to get an application health index. The traffic index mainly reflects the accessibility of the application, which is measured by the lost packets; the quality index mainly reflects the effectiveness of the application, which is measured by the delay and jitter data.

Multi-dimensional correlation analysis: The network O & M AI collector collects various types of data, mainly including the operating and transmission status of networking devices, user terminals, application flows, as well as the service and policy configuration. Then, the intrinsic relationship between the data is integrated to form multi-dimensional correlations [9,10], which help to provide timely warnings and detect problems and faults in the network from the following aspects:

Alarms of network switches, such as anomalies detected on device ports/optical modules:

Analyze which user-side servers and northbound or southbound service streams will be affected by the anomalies.

Alerts for control plane protocols, such as the connection exceptions of BGP neighbors: Analyze which service between these neighbors will be affected by the exception, and the service access of which tenants' servers will be affected.

Algorithms such as time-series data decomposition and machine learning can support anomaly analysis and dynamic prediction of network metrics [11–13]. Specifically, the anomaly analysis is conducted for detecting anomalies in the past, while the dynamic prediction is conducted for predicting the future trend of data and getting a predicted graph. For the anomaly analysis, a dynamic baseline approach is adopted to model and train the historical data collected by devices, so as to form a dynamic prediction baseline as an anomaly detection threshold. The baseline can be improved through continuous automatic learning and iterative self-correction based on the historical data over a certain period [14,15]. The anomaly detection algorithm based on the dynamic baseline can reflect the actual network operating situation more accurately.

Table 2. Table of data metrics used to establish a baseline.

Category	Baseline Metrics	Longest Historical Data Training Period	Baseline Computing Period	Interval for Keeping Baseline Data
Device/Single board	CPU utilization rate Memory utilization rate	Almost 14 days	1 day	1 month
Interface	Number of packets received or sent	Almost 14 days	1 day	1 month
	Number of error packets received or sent			
	Number of lost packets received or sent			
	Number of broadcast packets received or sent			
Optical module	Received/sent power, current, voltage, and temperature of the optical module	Almost 14 days	1 day	1 month

The dynamic baseline is calculated offline every other day based on 14 days of historical data, that is, the predicted baseline value for the next day is calculated the day before at a time. The granularity of the generated dynamic baseline data is consistent with that of the original data.

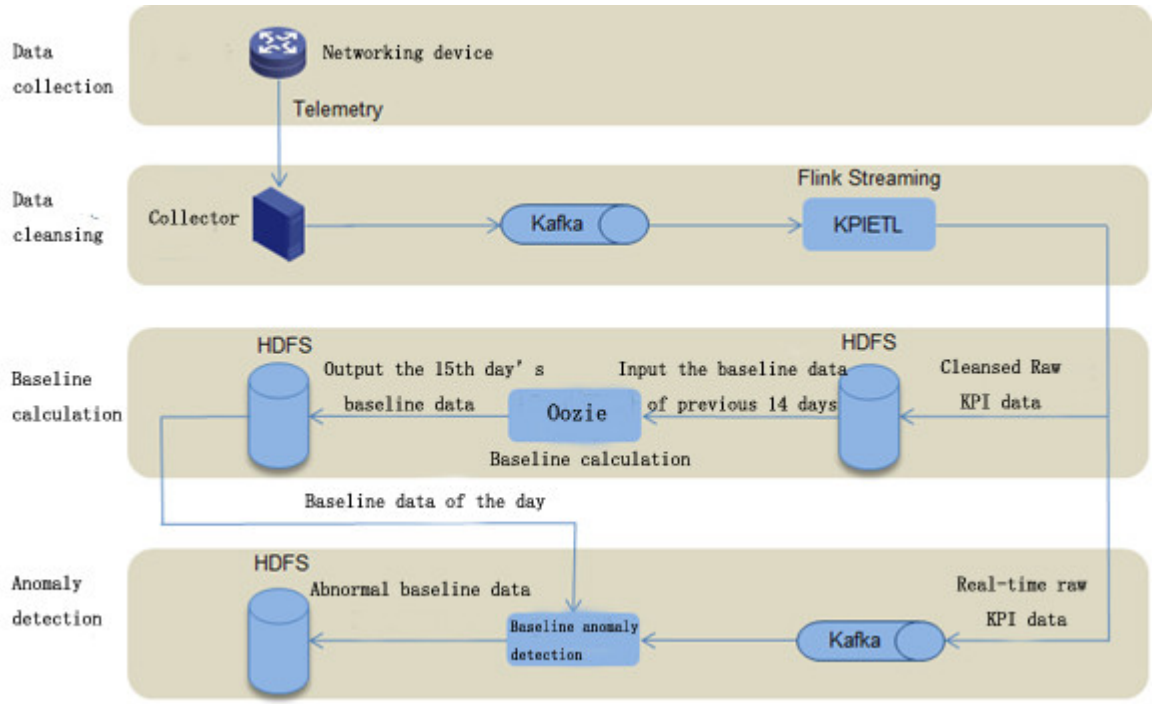

Figure 6. Dynamic baseline data streams.

Traditional network traffic management mainly stays at the level of equipment operation status. It can only judge the network carrying effect and capacity through the working status of equipment. This way can not reflect the real transmission status of the network and can not fully perceive the overall service carrying situation. Artificial intelligence network traffic analysis can fully perceive the service carrying traffic of the data center from a global perspective, and form a tightly coupled association model between network and business through business end-to-end visual modeling, so as to provide comprehensive technical support for optimizing network capacity and equipment risk early warning.

With the help of artificial intelligence technology, through machine learning and deep learning algorithms [16], the network becomes more flexible, observes the network from the perspective of application, and actively senses the problems existing in the network and application. It also provides automatic troubleshooting capability for business problems, quickly delimits and recovers faults, and reduces network operation and maintenance costs.

The artificial intelligence-based tuning strategy refers to the process of analyzing the ERSPAN-mirrored stream data, i.e., the INT stream data, and telemetry performance metrics from the network, application, and user dimensions, according to the actual application scenario of the data center network. In addition, AI algorithms such as anomaly detection dynamic baseline are used to perform intelligent analysis, which can proactively perceive whether there is a potential risk in the network and provide early warning. Network analysis is mainly to conduct dynamic real-time monitoring on the network topology, network devices, and device resources, judging whether errors occur through intelligent analysis and carrying out early warning. Examples are network link traffic monitoring, optical module monitoring, device CPU and memory utilization monitoring, chip resource monitoring at the forwarding level, etc. Application analysis focuses on identifying whether the interaction with the identified application is abnormal and whether the service quality of the application is abnormal,

including TCP anomaly detection, visualization and delay analysis of application forwarding paths. User analysis is concentrated on user experience, including user access abnormalities or failures. For example, baseline analysis and anomaly detection are performed by using the ratio of the number of users who have encountered access failures to the total number of associated users over a period of time as a KPI.

4. Conclusions

Automatic network strategy optimization based on artificial intelligence enable engineers to comprehensively perceive network traffic conditions, data response time, service transmission status and other information through automatic learning and data analysis, and automatically tune the network based on changes in network traffic and health status. In addition, consistency and integrity checks of the tuning strategy are automatically performed to reduce the chance of errors and the risk of network operation.

In the future, it will be promoted and applied in data centers at all levels to improve the accuracy of traffic analysis of various services in the data center. Realize the transformation from equipment configuration command to equipment configuration intention, speed up the service opening speed, shorten the batch configuration time, and reduce the pressure of operation and maintenance.

Acknowledgments

We thank the Northeast Branch of State Grid Corporation of China for its technical assistance and support.

Conflict of interest

The authors declare that they have no competing interest.

References

1. S. Smoot, Private cloud computing: Integration, virtualization and service-oriented infrastructure, Mechanical Industry Press, 2013.
2. Q. Wang, TCP/IP protocol analysis, *J. North. Jiaotong Univ.*, **1** (1995), 112–117. doi: CNKI:SUN:BFJT.0.1995-01-025.
3. Z. Y. He, O. Li, B. W. Yang, Y. Liu, Formal description of TCP protocol based on timed Colored Petri Net, *Comput. Eng.*, **37** (2011), 77–80. doi: CNKI:SUN:JSJC.0.2011-18-028.
4. C. Yang, Y. Q. Gui, Network traffic analysis and control strategies, *Comput. Knowl. Technol.*, **10** (2011), 26–28+62. doi: CNKI:SUN:BGDH.0.2011-10-012.
5. H. L. Zhang, H. Wang, A sFlow-based network traffic analysis method, *Comput. Eng. Sci.*, **8** (2007), 61–63+73. doi: CNKI:SUN:JSJK.0.2007-08-020.
6. R. Lv, W. Dai, G. Cao, Research on multiple combinations of network traffic analysis methods, *Small Microcomput. Syst.*, **31** (2010), 631–634.
7. L. Chen, B. Dong, X. Wang, Design and implementation of a global network traffic analysis system based on SNMP protocol, *J. Dalian Univ. Technol.*, **45** (2005), 69–72.

8. M. Xia, Research and application of root cause analysis of abnormal behavior of ethernet networks, *Liaoning Univ. Eng. Technol*., 2019.
9. G. Zhang, *Design and Implementation of SNMP-based Performance Monitoring System*, Ph.D thesis, Beijing Jiaotong University, 2008.
10. X. Li, *Design and Implementation of SNMP-based Network Device Performance Monitoring System,* Ph.D thesis, Beijing University, 2009.
11. X. Zhang, J. You, A gated dilated causal convolution based encoder-decoder for network traffic forecasting, *IEEE Access*, **8** (2020), 6087–6097. doi: 10.1109/ACCESS.2019.2963449.
12. B. Pfülb, C. Hardegen, A. Gepperth, S. Rieger, A study of deep learning for network traffic data forecasting, in *International Conference on Artificial Neural Networks*, (2019), 497–512. doi: 10.1007/978-3-030-30490-4_40.
13. S. H. Mousavi, M. Khansari, R. Rahmani, A fully scalable big data framework for botnet detection based on network traffic analysis, *Inf. Sci.*, **512** (2020), 629–640. doi: 10.1016/j.ins.2019.10.018.
14. M. Kim, H. Kong, S. Hong, S. Chung, J. W. Hong, A flow-based method for abnormal network traffic detection, in *2004 IEEE/IFIP network operations and management symposium (IEEE Cat. No. 04CH37507)*, **1** (2004), 599–612.
15. E. Corchado, Á. Herrero, Neural visualization of network traffic data for intrusion detection, *Appl. Soft Comput*., **11** (2011), 2042–2056. doi: 10.1016/j.asoc.2010.07.002.
16. T. J. M. Bench-Capon, P. E. Dunne, Argumentation in artificial intelligence, *Artif. Intell.*, **171** (2007), 619–641. doi: 10.1016/j.artint.2007.05.001.

AIMS Press