PROPOSED COMPREHENSIVE EAVESDROPPING DETECTION SOLUTION IN MULTI-VLAN SWITCHED NETWORK BASED ON IPFIX FLOW

Viet H. Le [1], Huy-Trung Nguyen [2], Cuong V. Trinh [1]
and Tran Minh Hieu [2]

[1] Department of Cybersecurity and High-Tech Crime Prevention,
People's Security Academy, Hanoi, Vietnam
[2] Research Institute of Posts and Telecommunications, Data Governance Laboratory,
Posts and Telecommunication Institute of Technology, Hanoi, Vietnam

ABSTRACT

With the rapid development of the Internet of Things trend, the components participating in traditional computer networks are increasingly complex, and the risk of eavesdropping is increasing and difficult to detect. In this paper, an eavesdropping detection method for multi-VLAN switching networks based on IPFIX flows is presented. The proposed method will generate and send special packets to stimulate responses from potential sniffers in VLAN networks. Combined with the use of IPFIX to collect and analyse flow data from network devices, it helps detect eavesdropping devices in VLAN networks. The proposed method has been proven effective in the EVE-NG simulation environment with different test scenarios.

KEYWORDS

Eavesdropping Detection; Multi-VLAN; IPFIX Flow.

1. INTRODUCTION

In the context of increasingly complex and large-scale network systems, ensuring network security becomes a significant challenge. Sniffing is one of the common threats, where attackers secretly collect data transmitted over the network without permission. This behaviour can lead to the theft of sensitive information, such as personal data, business secrets, or organisational information, resulting in severe financial, reputational, and security consequences.

Sniffing in switched networks, especially networks divided into multiple VLANs (Virtual Local Area Networks), is often more challenging to detect due to the segmented nature and complex data flows [1]. Standard eavesdropping techniques include [2], [3], [4]:

- ARP spoofing attack: Attackers spoof MAC addresses to redirect network traffic to their devices.
- Network interface sniffing: Use tools like Wireshark to capture packets on unencrypted network segments.
- Man-in-the-middle (MITM) attack: An attacker inserts himself between two communicating parties to eavesdrop or modify data.

In networks using VLANs, detecting eavesdropping becomes more challenging because the data traffic is logically separated; however, there is still a risk of exploitation if the VLAN

configuration is not strict enough or if vulnerabilities exist in the routing protocols. Traditional eavesdropping detection methods, such as pattern-based traffic monitoring or behavioural analysis, often struggle to handle large volumes of data and detect sophisticated attack patterns.

To solve this problem, this paper proposes an eavesdropping attack detection method using IPFIX (IP Flow Information Export) flows. IPFIX provides the ability to collect and analyse highly detailed network flow information, including source, destination, protocol, and other characteristics of packets. Based on this data, eavesdropping detection systems can analyse behaviour to identify abnormalities, such as an unusual data flow sent to an unwanted destination or a sudden increase in traffic on a specific port.

In sniffer detection, standard methods include: the ping method, ARP method, DNS method, source-route method, decoy method, and TDR method [5]. These methods are based on detecting abnormal responses from devices, such as ICMP or ARP responses from non-authentic devices. However, each method has its weaknesses, such as the ability to detect passive sniffers or false alarms, which requires them to be combined with additional methods to achieve higher efficiency in complex network environments. The purpose of this paper is to propose and present a sniffer detection method in an enterprise network environment that utilises the IPFIX flow monitoring mechanism, combined with packets that provoke passive sniffers to respond. This method aims to detect devices running in promiscuous mode, a standard method used by eavesdroppers to collect data in the network without being detected.

The primary contribution of this paper is to develop and test a novel method for detecting eavesdroppers in multi-VLAN networks, where the use of traditional monitoring tools is challenging. The paper provides experimental results demonstrating the ability to accurately detect devices in promiscuous mode without generating false alarms. At the same time, this method also ensures scalability and applicability in large-scale network environments with many VLANs and terminals.

In addition to the introduction, the paper is divided into the following sections: Section II Related works, presents related studies along with the limitations of previous related studies, thereby proposing an effective detection method; Section III Proposed method, a new proposed method to solve the problem of detecting eavesdropping attacks in complex enterprise network environments; Section IV Experimental implementation and evaluation, describes the testing process in a real network environment, test scenarios and results achieved. Experimental implementation with the proposed method and evaluates the effectiveness of the technique, at the same time pointing out the limitations and suggesting directions for future improvement; Section V Conclusion, summarises the main results, affirms the feasibility of the method for detecting eavesdropping devices in complex enterprise network environments, and suggests some future work.

2. RELATED WORKS

The following is a summary of the primary methods from related works. Each technique has specific weaknesses, especially in switched networks with many VLANs:

- Ping method [5]: It relies on responses from incorrect MAC addresses, but advanced sniffers can circumvent this by using virtual MAC filters. Additionally, Windows drivers can cause false positives due to the way multicast processing is handled. Not effective across VLANs without additional configuration.

- DNS Reverse Lookup [5]: Only effective with sniffers generating DNS traffic, but passive sniffers that do not do DNS lookups will not be detected and are limited in switched networks due to traffic fragmentation.
- ARP method [6]: It depends on local networks and is not effective in distributed VLANs. Sniffers can evade responses and generate large volumes, negatively impacting performance.
- ARP Cache Poisoning [7], [8]: Effective in broadcasts, but depends on promiscuous, not practical with switched networks, and generates large volumes, not suitable for large VLANs.
- IP Packet Routing [9]: Requires a monitoring port but necessitates scanning the entire network, generating significant traffic, and is limited to VLANs.
- Latency method [10]: According to the FAQ, it can reduce network performance and cause false positives due to network load, making it not practical for real-time monitoring in VLANs.
- Host method [5]: Requires direct access, is easily hidden by hackers, and is not scalable to VLAN networks.
- TDR and Hub Lights [5]: These methods are outdated and rarely used in modern switch networks due to their reliance on a star topology and lack of automation.
- SNMP Monitoring [5]: Effective only for hubs, it does not apply to switches, requires specialised hardware, and reduces applicability in VLANs.

The common points are limited scalability, high false positives, and ineffectiveness against passive sniffers, especially in VLAN networks. Based on the limitations, the proposed method for multi-VLAN switched networks has the following main components:

- Stimulus packet modulation: Create and send special packets (e.g., fake MAC/IP) to stimulate responses from potential sniffers. The packets are sent over trunk links that cover the entire VLAN, such as from core switches to hierarchical switches.
- IPFIX flow monitoring: Utilise IPFIX to collect and analyse flow data from network devices (such as switches and routers) to monitor responses. IPFIX provides detailed logs, including source and destination IP addresses, MAC addresses, and protocols, which help detect abnormalities such as unwanted traffic between VLANs.

This method of modulating packets ensures the responsiveness of passive sniffers across VLANs while leveraging the detailed monitoring capabilities of IPFIX, thereby solving the scalability problem associated with centralised analysis and reducing the need for per-VLAN deployment.

3. PROPOSED METHOD

3.1. Overview of Proposed Method

In the proposed method, the system is designed with the main components illustrated in Figure 1. In the first phase, the Packet Generator component is responsible for generating unusual packets. These packets are constructed with specific parameters, such as particular destination MAC addresses, custom VLAN tags, or unusual header fields, to stimulate responses from devices running in promiscuous mode.

After generating unusual packets, the system sends these packets to all hosts in the LAN, covering all VLANs. This is a critical step to ensure that all devices in the network receive this packet. Devices operating normally will ignore the packet because the NIC filters of modern operating systems or advanced passive sniffers filter packets with unusual MAC addresses. In

contrast, devices running sniffers in promiscuous mode can capture packets and generate responses.

To monitor all network activities, the system utilises an IPFIX Exporter combined with an IPFIX Collector. Network devices or sensors will export network flow data through the IPFIX Exporter. This data stream includes all information about packets and responses on the network, which is received and stored by the IPFIX Collector for analysis. This is a crucial clue for collecting response data from suspicious devices.

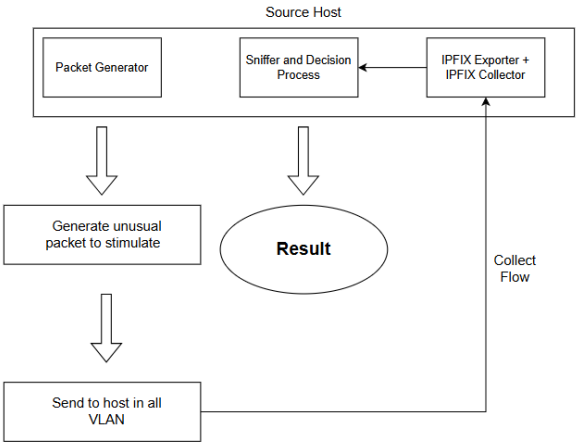

Fig. 1 Flow of the proposed method

Next, the Sniffer and Decision Process component will analyse the data streams collected from the IPFIX Collector. The analysis process involves determining whether any devices have responded to unusual packets. If a response from a device is detected to a packet that it should not have received, that device will be marked as a suspect running a sniffer.

Finally, the system will output the assessment results. This is the basis for determining which machines in the network are likely to be running in promiscuous mode.

3.2. Main Components of the Proposed Method

In this paper, we propose a method to detect sniffer-running devices (operating in promiscuous mode) by modulating anomalous ICMP Echo Request packets, utilising MAC address spoofing and monitoring network traffic using IPFIX. Still, the traditional ARP technique has been extended and improved to enhance detection capabilities in modern network environments, particularly in complex network systems with VLAN separation.

The traditional ICMP method, when using incorrect destination MAC addresses, is often ineffective in detecting promiscuous mode. The primary reason is that modern operating systems, along with network card drivers, are designed to eliminate packets with inappropriate destination MAC addresses at the hardware layer (NIC hardware filtering) or reject them by the operating system's software filter. Therefore, ICMP Echo Request packets with incorrect MAC addresses will usually not reach the software layer and cannot stimulate the device to respond.

However, through research and practical testing, we found that if an ICMP Echo Request packet is sent with a special destination MAC address (e.g, FF:FF:FF:FF:FF:FE) [6], the packet is not entirely discarded by the operating system or NIC. Instead, in specific network or operating system configurations, the packet may still be passed to the software layer for processing (due to

overlapping with multicast or variant broadcast receive logic). If the device is in promiscuous mode, it can process the packet and generate an ICMP Echo Reply.

Another difference in the paper's method is that the modulated ICMP packet structure has an unusual form:

- Ethernet Header: uses a special simulated destination MAC address (e.g., FF:FF:FF:FF:FF:FE), not a valid destination MAC address.
- VLAN Header: The packet is tagged with a VLAN tag (e.g., VLAN 10, VLAN 20) to ensure coverage of the entire VLAN domain within the enterprise network.
- IP Header: The destination IP address is the address of the suspect device.
- ICMP Header: Standard ICMP Echo Request, but embedded in a frame with an unusual MAC and VLAN.
- Payload: test data (e.g. "Hello"), which can have a non-standard size or content to increase the possibility of stimulating a response.

According to this unusual ICMP packet structure, the method can bypass the standard filters of the operating system on the target device and stimulate the device to respond when it is in promiscuous mode.

The reason this paper proposes the use of the IPFIX flow monitoring method is that it is beneficial in VLAN-segmented network environments, where devices in different VLANs do not respond directly when receiving an ICMP packet with an abnormal structure. Instead, anomalies can only be detected by monitoring IPFIX. IPFIX provides detailed information about data flows in the network, including the monitoring of anomalous packets without requiring direct responses. This is a powerful tool for identifying devices operating in promiscuous mode, because invalid or anomalous packets may not be returned immediately, but will appear in the flow data collected by IPFIX. By combining the method of modulating packets to provoke a response from the sniffer with IPFIX monitoring, our approach can detect and monitor promiscuous mode without requiring a direct response from the device. This is a significant improvement in detecting eavesdropping attacks in environments with VLAN partitions and modern network protection mechanisms.

With the theoretical basis for stimulating the machine running the eavesdropper to respond as mentioned above, we will present the specific workflow of the proposed method, illustrated in Figure 2, specifically including:

+**) Step 1: Generate** stimulus **packets**

The Packet Generator component generates ICMP Echo Request packets with the following characteristics:

- Ethernet Header: Special fake destination MAC, such as FF:FF:FF:FF:FF:FE.
- VLAN Header: The packet is tagged with a VLAN tag corresponding to the VLAN to which the target IP address belongs.
- IP Header: The destination IP address is all valid IP addresses currently active in each VLAN. The packet is not broadcast, but sent directly to each IP in the network.
- ICMP Header: Standard ICMP Echo Request.
- Payload: Test data that can be customised to create additional anomalies.

This packet structure modulation ensures that the ICMP packet is both standard at the IP layer and "different" enough at the MAC layer to stimulate a device in promiscuous mode that a standard device (not running an eavesdropping program) will ignore.

+) Step 2: Send packets to all IP addresses in the network

Packets are sent to all active IP addresses in all VLANs. This is the key point of the proposed method. Specifically, the modulated packet will not be sent in the usual broadcast or multicast manner. Each packet is explicitly sent to a valid IP but with a special destination MAC attached to it to bypass the filters of the operating system or the network switch. For standard devices, the NIC and the operating system will ignore it because the destination MAC address does not match. For devices in promiscuous mode due to running a sniffer, this device will receive the packet, process it and can respond with an ICMP Echo Reply to the “source host” address.

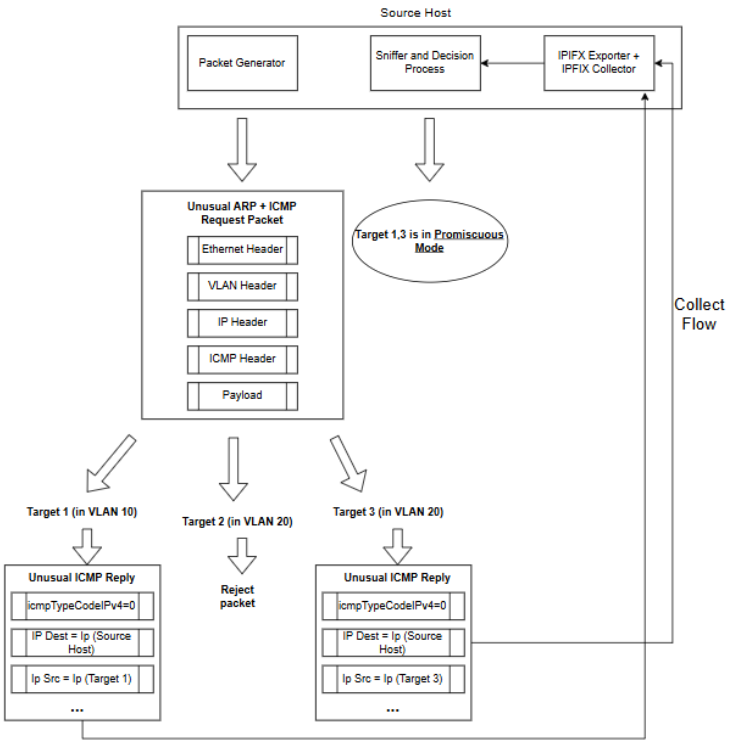

Fig. 2 Detailed workflow of the detection method

+) Step 3: Collect and monitor responses

On the device with the “source host” address, run the IPFIX Exporter configuration sniffing detection solution and IPFIX Collector to monitor the entire network. When there is a response after sending the stimulus packet, the corresponding “ICMP Echo Reply” responses from the devices running the sniffer will be recorded through the monitoring system. Specifically, the IPFIX Exporter on the “source host” will export the response stream information. IPFIX Collector collects data, recording details about the source IP, destination, ICMP type, source MAC address, and VLAN ID.

+) Step 4: Analyse the data stream and identify the suspect device

The Sniffer and Decision Process component performs analysis of the captured IPFIX data stream. A device is identified as a suspected eavesdropper if it satisfies one of the following conditions:

- If a device responds to an ICMP Echo Reply for a packet that it should not receive (destination MAC is not its own), the device is marked as running an eavesdropper.
- Other characteristics: icmpTypeCodeIPv4 has unusual values in the log stream output by the IPFIX protocol, the destination IP responds correctly to the IP of the "source host", but the source IP is that of the device that should not respond.

+) Step 5: Output the results of identifying the eavesdropper

The system and output compile the results of the analysis and identification of the eavesdropper as a report with the following information:

- List of information such as IP address, MAC, and VLAN ID of the suspected device.
- A message warning that the device is running a real-time eavesdropper.

4. EVALUATION

4.1. Environment Deployment

The evaluation environment was deployed on a virtualised server using the EVE-NG (Emulated Virtual Environment Next Generation) platform. EVE-NG was chosen because it is a powerful and flexible solution that allows simulation of large-scale network infrastructure with devices and software that are nearly identical to the real environment. The simulation system was deployed on a physical server with a multi-core CPU configuration, 32 GB of RAM, and a 500 GB SSD to ensure processing performance during the simulation process.

4.1.1. Simulation Network System

The network system is designed according to the simulation diagram as shown in Figure 3, including:

- A Router device (3725): performs the routing function between VLANs and connects to the Internet (Cloud1).
- A Switch Core device (SW_Core): In the test model, the Switch Core not only performs the VLAN transmission function but also takes on the role of collecting data flow (flow data) to serve sniffer detection analysis. To do this, we configure additional port monitoring (SPAN port) or equivalent features (if the switch supports), to:

  + Copy all traffic going through the trunk of VLANs to a monitoring port (monitor port).
  + The monitor port output connects to the IPFIX receiver

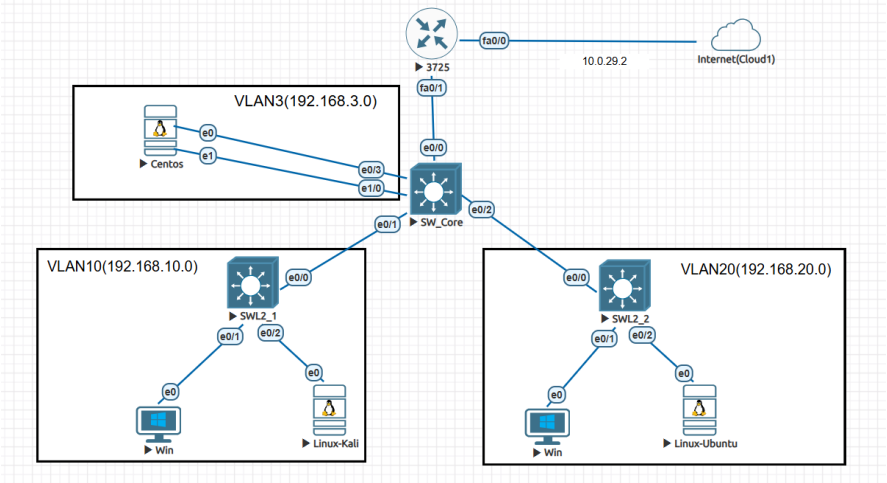

Fig. 3 Network system diagram for evaluation

- Two layer 2 Switch devices (SWL2_1, SWL2_2): connect terminals belonging to different VLANs.
- Terminals:

  + VLAN 3 (192.168.3.0/24): a CentOS server.
  + VLAN 10 (192.168.10.0/24): one Windows computer, one Linux Kali machine.
  + VLAN 20 (192.168.20.0/24): one Windows computer, one Linux Ubuntu machine.

These VLANs are routed through trunk ports to the layer two switch, and the endpoints are assigned the correct VLANs via the access port.

The EVE-NG simulation infrastructure replicates a standard enterprise network, with a router performing inter-VLAN routing, a core switch transporting and separating VLANs via trunk, and a layer two switch providing access connectivity to the endpoints. This model ensures the ability to deploy real-world tests for the eavesdropper detection method, including sending modulated challenge packets to all devices in the network and collecting the response stream.

4.1.2. Configure IPFIX Stream Monitoring and Collection in the Evaluation System

In the experimental system, to serve the purpose of monitoring the entire network traffic and detecting devices operating in promiscuous mode, the paper deploys a monitoring architecture based on the SPAN port mechanism combined with the IPFIX flow collection and export system. The designed configuration ensures comprehensiveness, objectivity and compatibility with the actual enterprise network infrastructure.

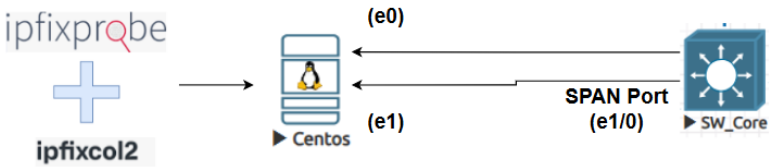

Fig. 4 IPFIX receiver configuration

On the Switch Core (SW_Core) device, all traffic through the trunk ports connecting the lower-level devices and routers is configured to mirror to a monitor port. The SPAN port mechanism accurately copies all traffic passing through the trunk ports without affecting the main data flow in the network. The mirrored traffic is transmitted to the Linux CentOS server in the core network area for analysis. On the Linux CentOS server, an additional network interface is explicitly configured for collecting IPFIX flows. The remaining network interface is typically connected to the network system. The network tools and configuration for capturing IPFIX streams are illustrated in Figure 4.

At the server receiving the mirror port, the article deploys the IPFIXprobe tool (Fig 5). In the system, ipfixprobe acts as an IPFIX exporter used to collect raw traffic from the monitor port (SPAN), collect packets, extract necessary indicators (such as ICMP type/code, IP address, MAC address) and export data as IPFIX flows. This IPFIX stream is sent via the UDP protocol to a local collector on the server for storage and further processing.

IPFIXcol2 is deployed as an IPFIX collector, responsible for receiving the IPFIX stream output by IPFIXprobe. IPFIXcol2 is configured to log data in JSON format, stored in a date-month-year directory structure, serving the post-analysis process and ensuring easy management and retrieval (Fig 6). Data is stored in periodic rotation sessions (e.g. a file every 5 minutes), optimising storage capacity and ensuring continuity in monitoring.

```
[user@localhost ~]$ sudo /bin/ipfixprobe -i "raw;ifc=ens4" -p dns -o "ipfix;host=127.0.0.1;port=4739;udp"
^CInput stats:
  #      packets       parsed              bytes      dropped           qtime status
  0           99           97              29124            0        12000043     ok
SUM           99           97              29124            0        12000043

Output stats:
  #      biflows      packets          bytes (L4)      dropped status
  0           27           97               27766            0     ok
[user@localhost ~]$
```

Fig. 5 Configuration using ipfixprbe

4.1.3. Preprocessing and Anomaly Detection

The captured IPFIX flow includes detailed information fields such as protocol, source and destination IP addresses, source and destination MAC addresses, ICMP parameters (icmpTypeCodeIPv4), TCP flags (if present), time, and other network session characteristics.

During the analysis, the system focuses on detecting ICMP Echo Reply flows (icmpTypeCodeIPv4 = 0) from devices that are not the actual destination of the ICMP Echo Request packet sent. These flows are identified as anomalous because:

- The ICMP Echo Request packet was initially sent to a valid destination IP address on the network but attached to a destination MAC address that could bypass the operating system's NIC filter or not be discarded by switches (e.g. FF:FF:FF:FF:FF:FE).
- A normally functioning device would discard the packet because the destination MAC address does not match the actual MAC of the device.
- However, a device in promiscuous mode will accept all traffic, including ICMP packets with special destination MACs. The device's software layer processes the packet and responds with an ICMP Echo Reply, creating an abnormal flow that the collector detects.

The main reason for the appearance of abnormal ICMP flows is that the destination device is enabling promiscuous mode on the network card. When in this mode, the device does not perform packet filtering based on MAC addresses at the hardware layer (NIC), but receives all

packets passing through, and transfers them to the software layer for processing. This causes the device to:

- Receive and process packets that do not have the same destination MAC address as their own MAC.
- Generate an ICMP Echo Reply response even though it is not, in principle, a valid destination device of the stimulus packet.

This response creates an IPFIX flow that is recorded by the collector system and becomes the basis for detecting devices running eavesdroppers or having unusual eavesdropping behaviour in the network as shown in Figure 7. The entire monitoring process is deployed in the following steps:

- Step 1: SW_Core configures the SPAN port to mirror all traffic from trunk ports to the monitoring port.
- Step 2: ipfixprobe (IPFIX exporter) collects mirror traffic, extracts the ICMP index, and exports the IPFIX stream.
- Step 3: ipfixcol2 (IPFIX collector) receives IPFIX stream, records JSON log for analysis.
- Step 4: The system post-processes JSON logs, detects abnormal ICMP Echo Reply streams, and identifies suspicious devices running promiscuous mode.

```
<ipfixcol2>
  <inputPlugins>
    <input>
      <name>UDP collector</name>
      <plugin>udp</plugin>
      <params>
        <localPort>4739</localPort>
      </params>
    </input>
  </inputPlugins>

  <outputPlugins>
    <output>
      <name>Simple JSON Output</name>
      <plugin>json</plugin>
      <params>
        <tcpFlags>formatted</tcpFlags>
        <timestamp>formatted</timestamp>
        <protocol>formatted</protocol>
        <ignoreUnknown>true</ignoreUnknown>
        <ignoreOptions>true</ignoreOptions>
        <nonPrintableChar>true</nonPrintableChar>
        <outputs>
          <file>
            <path>/tmp/ipfixcol/flow/%Y/%m/%d/</path>
            <prefix>json.</prefix>
            <timeWindow>300</timeWindow>
          </file>
        </outputs>
      </params>
    </output>
  </outputPlugins>
</ipfixcol2>
```

Fig. 6. Configure IPFIXcol2 to receive IPFIX

Fig. 7 Abnormal log line feedback from the listening device

4.2. Test scenarios and Results

To evaluate the effectiveness of the eavesdropping detection method based on the IPFIX flow monitoring mechanism, we built and implemented three test scenarios corresponding to everyday situations in practice. The test network system is designed with multiple VLANs, simulating a typical enterprise network environment with many terminals belonging to different subnets. In the scenarios, the system transmits ICMP Echo Request packets with destination MAC addresses that pass the operating system's NIC filter and the switch's MAC filtering mechanism to all valid IP addresses in the network, then records and analyzes responses from the terminals.

4.2.1. First Test Scenario

In the first scenario, all devices in the system operate in normal mode, without enabling promiscuous mode. The test results show that the system does not record any ICMP Echo Reply responses from devices that are not the real destination of the stimulus packet. The JSON logs collected from the IPFIX collector only record ICMP Echo Request streams that have been sent, with absolutely no unusual ICMP Echo Reply data (as shown in Fig 8). This is an important indication that the method does not generate false alarms in a secure network environment (as shown in Fig 9).

4.2.2. Second Test Scenario

In the second scenario, enable Wireshark on a device in the network (a Linux-Ubuntu machine in VLAN 10). When the system sends an ICMP Echo Request packet with a fake destination MAC, the promiscuously enabled device receives and processes the packet, generating an ICMP Echo Reply even though it is not the target device. The system correctly records this abnormal ICMP Echo Reply flow in the JSON log with complete information about the device. At the same time, the alert interface displays information about the suspect device (as shown in Fig 10).

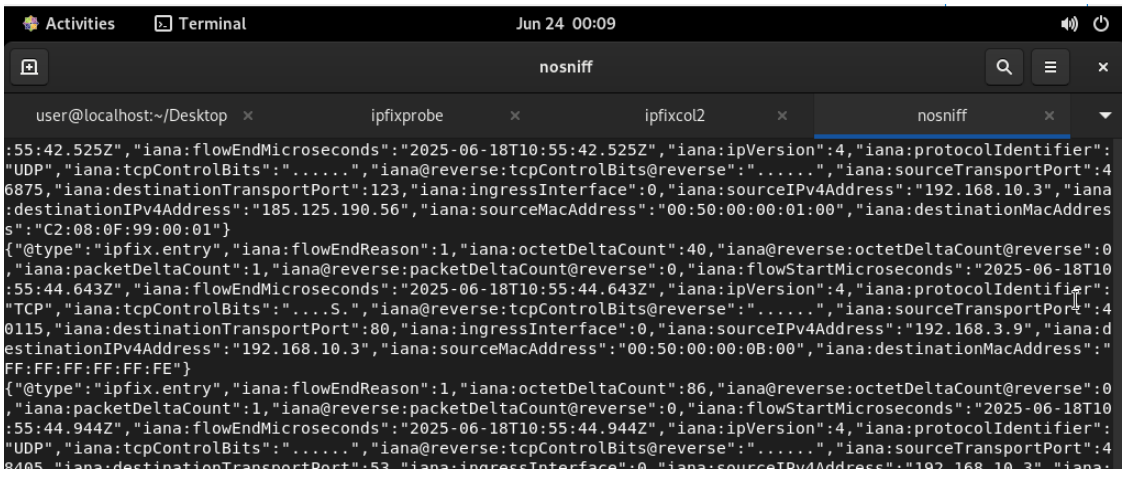

Fig. 8 Regular log lines represent modulated packets sent to the machine of interest

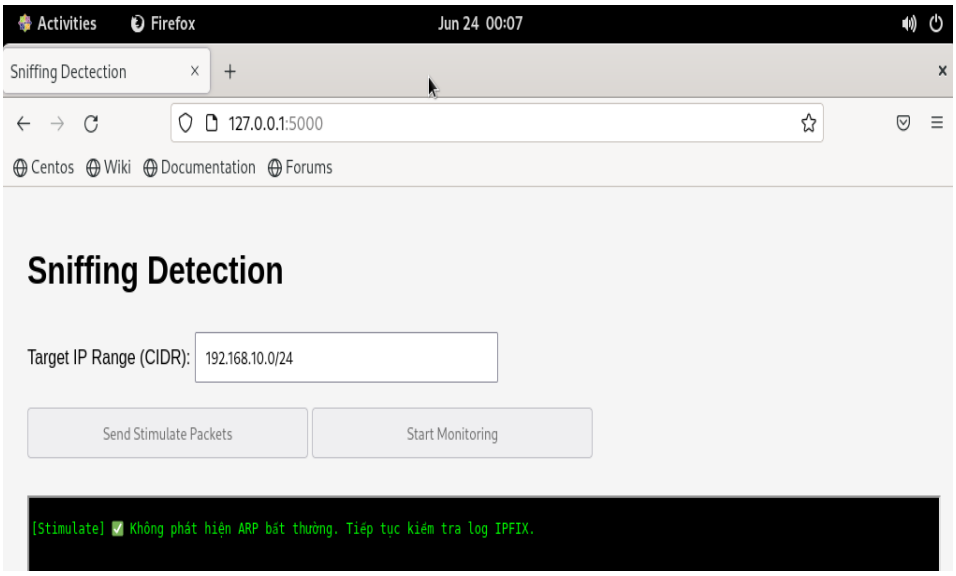

Fig. 9 Test results for the first scenario

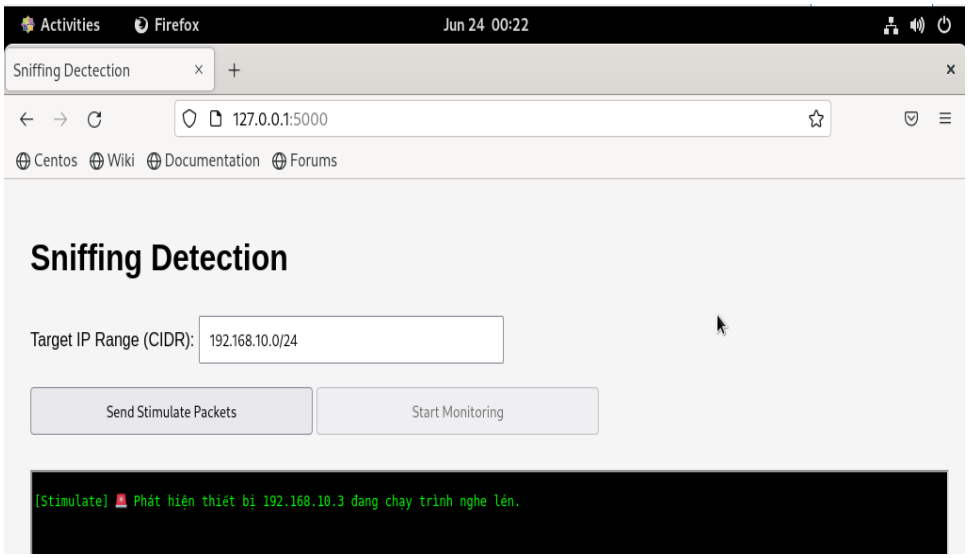

Fig. 10 Test results for the second scenario

4.2.3. Third Test Scenario

In the third scenario, two devices belonging to different VLANs are enabled in promiscuous mode (a Windows device in VLAN 20 and a Linux device in VLAN 10). The results show that the system simultaneously detects two abnormal ICMP Echo Reply flows originating from two suspicious devices. The centralised alert interface simultaneously displays two alerts that two devices belonging to two VLANs with different network ranges are running sniffers (as shown in Fig. 11).

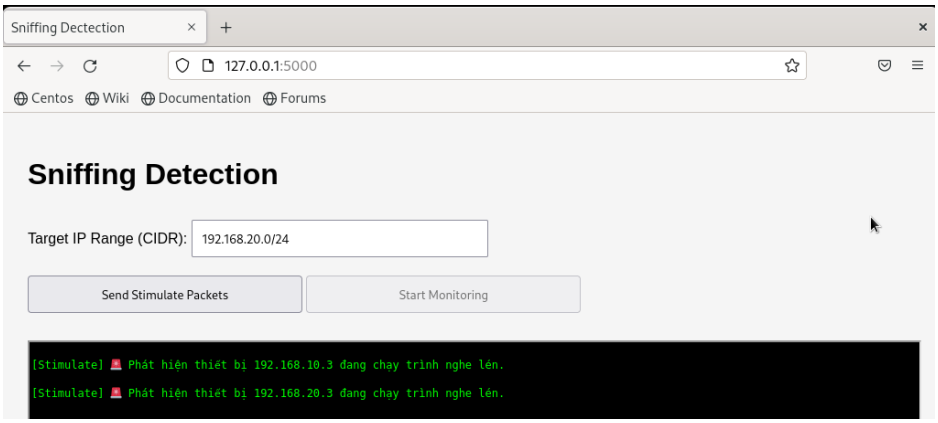

Fig. 11 Test results for the third scenario

Experimental results through scenarios show that the proposed method is capable of accurately detecting devices running eavesdroppers without generating false alarms when the system is operating normally. The technique also ensures scalability when applied in multi-VLAN network environments with a large number of devices.

5. DISCUSSION

5.1. Detection Capability

The method of using ICMP Echo Request packets with unusual MAC addresses and VLANs combined with monitoring via IPFIX is capable of detecting promiscuous mode effectively in VLAN-segmented network environments. This method bypasses traditional hardware and software filters, thanks to the use of special destination MAC addresses (FF:FF:FF:FF:FF:FE) and invalid VLAN tags. ICMP Echo Request packets may not be discarded entirely and may be processed by devices in promiscuous mode. This creates an opportunity for the device to respond with an ICMP Echo Reply, which helps identify devices in this mode.

However, it should be noted that this method does not provide immediate feedback from the device. ICMP packets may only be recorded in the flow data collected by IPFIX, instead of receiving an ICMP reply immediately. IPFIX monitoring helps identify anomalies in network data streams, creating an opportunity to detect devices in promiscuous mode, without requiring direct feedback from the device.

Based on the ability to bypass conventional filters and monitor across network data streams, this method can detect suspicious devices with high accuracy in segmented VLAN networks, where devices often do not directly respond to provocation packets.

5.2. Weaknesses and Shortcomings of the Method that have not been Overcome

- Dependent on network configuration and operating system: This method does not always guarantee that ICMP Echo Request packets will be processed at the software layer, because in some network configurations or operating systems, these packets may be discarded entirely at the hardware layer or software filters before being processed. This may reduce the effectiveness of the method in some network environments, especially when the device or network is configured to protect against invalid packets

- Dependence on IPFIX monitoring: This method requires monitoring via IPFIX to detect anomalies. Without an IPFIX monitoring tool or if IPFIX is not deployed correctly throughout the network, the detection capability may be reduced. Furthermore, collecting and analysing data from IPFIX requires significant processing resources, which may affect the performance of the system when monitoring large-scale networks

- Applicability in real environments: Although this method can detect promiscuous mode in theory and experiments, applying this method in real networks may encounter some problems, such as complex network configurations, constant changes in operating systems and network drivers, or strict network security and control measures.

6. CONCLUSIONS

The paper presents a new method for detecting promiscuous mode in network devices through the use of ICMP Echo Request packets with unusual MAC addresses and VLANs. This method successfully bypasses the usual filters of operating systems and drivers, allowing ICMP Echo Request packets to be processed at the software layer and triggering ICMP Echo Reply responses when the device is in promiscuous mode. Thanks to the combination of techniques, such as using special MAC addresses and invalid VLAN tags, the method is capable of detecting devices in promiscuous mode without requiring immediate direct response. A special feature of the process is the use of IPFIX monitoring to detect anomalies in network data flows, helping to identify suspicious devices without requiring immediate response from the device. This approach is practical in segmented VLAN networks where devices often do not respond directly, but can still be detected through network monitoring.

Although this approach has achieved promising results, there are still some limitations that need to be addressed. Future research directions will focus on the following:

- Improving detection in complex network environments: An important research direction is to enhance detection in complex networks with multiple layers of protection and strong security configurations. ICMP packet optimisation and monitoring through network tools such as IPFIX can be improved to detect promiscuous devices in environments with multiple protection measures. Developing detection methods combined with other security technologies: Research can be extended to combining this detection method with other security technologies such as IDS/IPS (Intrusion Detection/Prevention Systems), network behaviour analysis systems, and network access control measures to enhance the ability to detect and prevent potential threats.

- Enhance the applicability in enterprise and large networks: A significant challenge is to apply this method in large enterprise networks with thousands of devices. Therefore, further research will aim to optimise the technique so that it can be effectively applied in large networks, minimising the processing load when monitoring and collecting data from IPFIX.

With these research directions, the promiscuous mode detection method can be enhanced, applied more widely, and effectively promoted in modern and highly secure network environments.

DISCLOSURE STATEMENT

No potential conflict of interest was reported by the author(s).

FUNDING

The authors received no financial support for the research, authorship, and/or publication of this article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] "Sniffing attacks on computer networks." Accessed: June 23, 2025. [Online]. Available: https://www.researchgate.net/publication/355513843_Sniffing_attacks_on_computer_networks

[2] J. of C. S. Ijcsis, "Network Sniffing And Its Consequences: A Comprehensive Survey," *International Journal of Computer Science and Information Security (IJCSIS), Vol. 22, No. 3, June 2024*, Jan. 2024, Accessed: June 23, 2025. [Online]. Available: https://www.academia.edu/121181248/Network_Sniffing_And_Its_Consequences_A_Comprehensive_Survey

[3] P. Anu and S. Vimala, "A survey on sniffing attacks on computer networks," in *2017 International Conference on Intelligent Computing and Control (I2C2)*, June 2017, pp. 1–5. doi: 10.1109/I2C2.2017.8321914.

[4] B. Prabadevi, N. Jeyanthi, N. I. Udzir, and D. Nagamalai, "Lattice structural analysis on sniffing to denial of service attacks," *International Journal of Computer Networks & Communications (IJCNC)*, vol. 11, no. 4, July 2019, doi: https://doi.org/10.5121/ijcnc.2019.11406.

[5] "Sniffing." Accessed: June 23, 2025. [Online]. Available: https://cs.baylor.edu/~donahoo/tools/sniffer/sniffingFAQ.htm

[6] D. Sanai, "Detection of Promiscuous Nodes Using ARP Packets".

[7] Z. Trabelsi and H. Rahmani, "Detection of Sniffers in an Ethernet Network," in *Information Security*, K. Zhang and Y. Zheng, Eds., Berlin, Heidelberg: Springer, 2004, pp. 170–182. doi: 10.1007/978-3-540-30144-8_15.

[8] A. N. Khan, K. Qureshi, and S. Khan, "An Intelligent Approach of Sniffer Detection".

[9] "(PDF) Switched Network Sniffers Detection Technique Based on IP Packet Routing," *ResearchGate*, Accessed: June 23, 2025. [Online]. Available: https://www.researchgate.net/publication/220449840_Switched_Network_Sniffers_Detection_Technique_Based_on_IP_Packet_Routing

[10] K. Cabaj, M. Gregorczyk, W. Mazurczyk, P. Nowakowski, and P. Żórawski, "Sniffing Detection within the Network: Revisiting Existing and Proposing Novel Approaches," in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, in ARES '19. New York, NY, USA: Association for Computing Machinery, Aug. 2019, pp. 1–8. doi: 10.1145/3339252.3341494.

AUTHORS

Viet H. Le received a PhD in Information Systems from the Graduate University of Science and Technology, VAST, Hanoi, Vietnam, in 2023. He is currently a Lecturer with the Department of Cybersecurity and High-Tech Crime Prevention, People's Security Academy, Hanoi, Vietnam. His work has involved research, application, and teaching in the fields of network security, artificial intelligence, machine learning, and malware analysis. He has many books and more than ten research articles to his credit.

Trung H. Nguyen earned the Master's degree in Information Technology from Hanoi University of Science and Technology (Vietnam) in 2017, followed by a Ph.D from the Graduate University of Science and Technology under the Vietnam Academy of Science and Technology in 2021. Currently, he serves as a Lecturer at the Research Institute of Posts and Telecommunications and holds the position of Head of the Data Governance Laboratory at the Posts and Telecommunication Institute of Technology in Hanoi, Vietnam. His expertise lies in advancing security technologies and solutions, with a focus on cultivating innovative approaches to information security that deliver practical value to both industry and academia. He has also served as a reviewer, and TPC Chair for several international conferences and journals.

Cuong V. Trinh is a student at the Department of Cyber Security and High-Tech Crime Prevention, People's Security Academy, Hanoi, Vietnam, since 2020. His academic interests focus on computer network security, packet analysis, network design, and artificial intelligence. He has received several awards in national Olympiads in Informatics, scientific research contests, and cybersecurity competitions. He has also authored and co-authored publications presented at both national and international conferences in the fields of cybersecurity, artificial intelligence, and computer networking.

Tran Minh Hieu received the Engineer degree in Information Technology from the Posts and Telecommunications Institute of Technology, followed by a Master's degree in Information Systems from the same institute in 2023. He is currently a Lecturer in Information Technology at the Research Institute of Posts and Telecommunications. His research interests focus on machine learning, the development and deployment of network systems, and information security.