Real-time Network Device Configuration and Security Monitoring System Using NLP and LLM

T. Suganya[1], M. Mohamed Apsal[2], L.V. Shriramsankar[3], B. Niranjan[4],
Assistant Professor[1], Students[234],
Department of Computer Science and Engineering(Cybersecurity),
K.L.N. College of Engineering, Pottapalayam, Sivagangai.

Abstract—Modern enterprise networks contain a wide range of devices, services, and security challenges, making traditional manual configuration difficult and prone to human error. To address this issue, this work proposes a natural language–based network automation and security monitoring system that simplifies device configuration and improves operational efficiency. In this system, network administrators can express high-level intents, such as enabling SSH access, configuring system logging, or checking device status, using simple natural language commands. These commands are processed using Natural Language Processing (NLP) techniques and Large Language Models (LLMs) to automatically generate the corresponding router configuration commands. The generated configurations are then applied within a simulated enterprise network environment for real-time device management. In addition to automation, the system continuously monitors network interfaces and device behavior to identify issues such as unauthorized port activity, interface failures, or unusual network events. When such conditions are detected, alerts are generated to notify administrators. By combining intent-based automation with real-time monitoring, the system reduces manual workload, decreases the likelihood of configuration errors, and improves overall network reliability and security. This solution demonstrates a practical and scalable approach for managing modern enterprise networks efficiently.

Keywords: Network Automation, Intent-Based Networking, Natural Language Processing (NLP), Netmiko, Network Security Monitoring, Intrusion Detection System (IDS)

I. INTRODUCTION

Today's organisation networks include a couple of routers, switches, and interconnected devices that require non-stop configuration and monitoring to ensure green operation and security. Historically, network directors configure these devices manually the use of command-line interfaces (CLI), which can be complicated, time-consuming, and vulnerable to human errors. As community length and complexity boom, guide configuration becomes inefficient and difficult to control, regularly leading to misconfigurations and protection vulnerabilities.

Latest advancements in artificial intelligence and herbal Language Processing (NLP) have enabled the improvement of clever structures that simplify complex technical obligations. In networking, motive-based totally tactics allow administrators to define excessive-degree necessities in natural language, which can be routinely translated into device-unique configuration instructions. This reduces the dependency on guide CLI operations and improves ordinary community management efficiency.

In this paper, an AI-based cause-pushed network automation and security tracking gadget is proposed. The system lets in users to enter network configuration intents in easy language, which are processed to generate corresponding router commands and deployed routinely the usage of SSH-based totally automation. in addition to configuration, the gadget constantly monitors community interfaces to stumble on unauthorized get right of entry to and interface screw ups. by way of combining automation with real-time tracking, the proposed system enhances network reliability, reduces administrative effort, and improves usual community security.

II. LITERATURE REVIEW / RELATED WORK

Recent research has focused on improving network management through automation, intent-based networking, and intelligent systems. Several studies have explored different approaches to simplify configuration processes and enhance network security.

INSpIRE: Integrated NFV-based intent refinement environment [1] proposed an intent-based framework that refines high-level user requirements into network configurations using Network Function Virtualization (NFV). The system focuses on translating user intent into actionable policies, improving flexibility in network management. However, it mainly emphasizes service orchestration rather than real-time monitoring.

A comprehensive approach to the automatic refinement and verification of access control policies [2] introduced a method for automating the refinement and verification of access control policies. Their approach enhances network security by ensuring correctness in policy implementation. While effective in policy validation, it does not address dynamic configuration or real-time device-level monitoring.

IBCS: Intent-based cloud services for security applications [3] presented an intent-based cloud service model designed for security applications. The system allows users to define security requirements at a higher level, which are then implemented automatically. Although it improves cloud security management, it is primarily focused on cloud environments rather than enterprise network devices.

Hey, Lumi! Using natural language for intent based network management [4] explored the use of natural language interfaces for network management. Their work demonstrates how user inputs in plain language can be translated into network configurations. This approach improves usability, but it lacks integration with continuous monitoring and alert mechanisms.

A survey on intent based networking [5] provided a comprehensive survey of intent-based networking technologies, highlighting their benefits, challenges, and future directions. The study emphasizes the importance of automation in modern networks but does not propose a complete implementation combining multiple functionalities.

Intent-driven autonomous network and service management in future cellular networks: A structured literature review [6] reviewed intent-driven network management approaches in next-generation cellular networks. The authors discussed the role of automation and intelligence in managing complex systems, but their focus is mainly on large-scale telecom infrastructures.

From the analysis of existing works, it is observed that most solutions focus on either intent-based automation or security aspects independently. Very few systems integrate natural language-based configuration, automated deployment, and real-time monitoring into a single framework. To address this gap, the proposed system combines NLP-based intent processing with automated configuration and continuous network monitoring, providing a more comprehensive and practical solution for modern enterprise networks.

III. PROPOSED SYSTEM

The proposed system is an intelligent network automation and security monitoring solution that integrates Natural Language Processing (NLP) with automated configuration and real-time monitoring. The primary objective of the system is to simplify network management by allowing administrators to interact with network devices using high-level natural language commands instead of manual command-line configuration.

In this system, the user provides input in the form of simple text instructions through a web-based interface. These instructions may include tasks such as enabling SSH access, configuring IP addresses, setting up routing protocols, or checking device status. The system processes the input using an intent analysis mechanism to identify the required network operation.

Once the intent is identified, the command generation module converts the user’s request into device-specific configuration commands. These commands are structured according to the syntax supported by network devices. The generated commands are then securely deployed to the target device using an SSH-based automation module, ensuring safe and reliable communication.

In addition to configuration automation, the system includes a continuous monitoring component that observes the status of network interfaces in real time. The monitoring module checks for abnormal conditions such as unauthorized interfaces becoming active, trusted interfaces going down, or unusual device behavior.

When such anomalies are detected, the system triggers an alert mechanism that notifies the administrator through email. This ensures that network issues are identified and addressed at an early stage, reducing the risk of failures and security threats.

The integration of natural language-based automation with real-time monitoring makes the proposed system efficient, user-friendly, and reliable. It significantly reduces manual effort, minimizes configuration errors, and enhances overall network security. Compared to existing systems, the proposed solution provides a unified approach by combining configuration, monitoring, and alerting within a single framework.

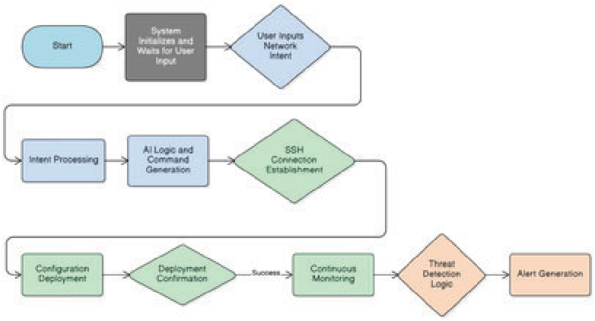


Fig 1. Data flow diagram of proposed system

IV. SYSTEM ARCHITECTURE

The proposed system follows a modular architecture that integrates user interaction, automation, and monitoring to manage network devices efficiently. Each module performs a specific function and collectively provides a complete network management solution.

The process begins with a web-based user interface, where the administrator provides instructions in natural language. These inputs are sent to the backend server for processing. The intent processing module analyzes the input and identifies the required network operation using predefined keywords or rules.

Based on the identified intent, the command generation module creates device-specific configuration commands. These commands are then executed on the target device through the deployment module, which establishes a secure SSH connection using automation tools such as Netmiko.

After configuration, the monitoring module continuously checks the status of network interfaces and device activity. The alert module detects abnormal conditions such as unauthorized access or interface failures and notifies the administrator through email.

This architecture enables automated configuration and real-time monitoring in a single system, improving efficiency, reducing errors, and enhancing network security.

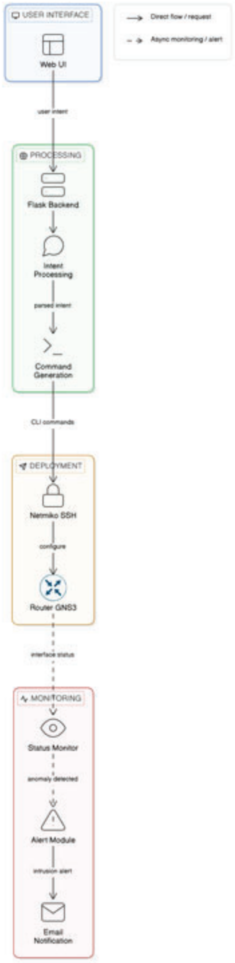


Fig 2. System architecture diagram

V. IMPLEMENTATION & METHODOLOGY

1. System Design Overview

The system is designed as a modular architecture that integrates user interaction, intent processing, command generation, configuration deployment, and network monitoring. Each module works independently but is connected to form a complete automated network management system.

2. Development Environment

The implementation is carried out using Python as the primary programming language due to its simplicity and strong support for network automation. A web interface is developed using the Flask framework to allow user interaction. The network environment is simulated using a router setup, enabling safe testing of configurations and monitoring features.

3. Intent Processing Mechanism

The system accepts user input in the form of natural language. The intent processing module analyzes the input using keyword-based logic to identify the required network operation. Based on the detected keywords such as “SSH”, “gateway”, or “OSPF”, the system determines the appropriate configuration task.

4. Command Generation Process

Once the intent is identified, the command generation module converts the user request into device-specific configuration commands. These commands are structured in the format supported by network devices, ensuring compatibility and correct execution.

5. Configuration Deployment

The generated commands are deployed to the network device using a secure SSH connection. The deployment module establishes communication with the router and sends the commands automatically. This eliminates the need for manual configuration and ensures consistent execution of network operations.

6. Network Monitoring Mechanism

After configuration, the system continuously monitors the network device by executing standard diagnostic commands. It retrieves interface status information and analyzes it to identify abnormal conditions such as inactive interfaces or unexpected activity.

7. Intrusion Detection Logic

The monitoring module uses a rule-based approach to detect anomalies. It compares active interfaces with a predefined list of authorized interfaces. If an unauthorized interface becomes active or a critical interface goes down, the system identifies it as a potential issue and generates an alert.

8. Alert Generation and Notification

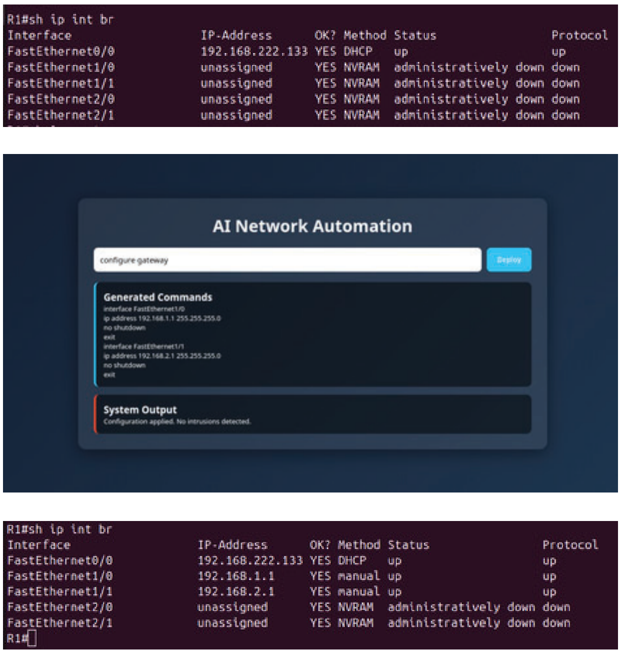
When an abnormal condition is detected, the system generates an alert message and notifies the administrator. The alert mechanism includes email notification using secure communication protocols, ensuring that the administrator is informed in real time.

9. User Interface Interaction

The web-based interface allows users to enter network intents and view system responses. After submitting an input, the user receives feedback such as generated commands, deployment status, and monitoring results, providing a simple and interactive experience.

10. Periodic Monitoring

In addition to manual checks, the system supports periodic monitoring at fixed time intervals. This ensures continuous observation of the network and helps in early detection of issues without requiring user intervention.



```
R1#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.222.133 YES DHCP   up                    up
FastEthernet1/0        unassigned      YES NVRAM  administratively down down
FastEthernet1/1        unassigned      YES NVRAM  administratively down down
FastEthernet2/0        unassigned      YES NVRAM  administratively down down
FastEthernet2/1        unassigned      YES NVRAM  administratively down down
```

AI Network Automation

configure gateway

Generated Commands

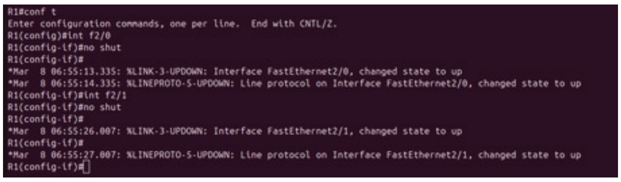
```
interface FastEthernet1/0
ip address 192.168.1.1 255.255.255.0
no shutdown
exit
interface FastEthernet1/1
ip address 192.168.2.1 255.255.255.0
no shutdown
exit
```

System Output

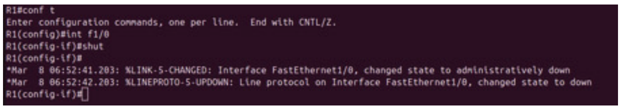
Configuration applied. No intrusions detected.

```
R1#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.222.133 YES DHCP   up                    up
FastEthernet1/0        192.168.1.1     YES manual up                    up
FastEthernet1/1        192.168.2.1     YES manual up                    up
FastEthernet2/0        unassigned      YES NVRAM  administratively down down
FastEthernet2/1        unassigned      YES NVRAM  administratively down down
R1#
```

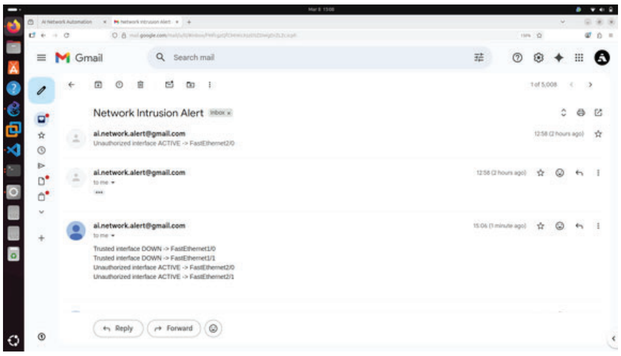
Fig 3. The gateway is configured via our system by giving prompt



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int f2/0
R1(config-if)#no shut
R1(config-if)#
*Mar  8 06:55:13.335: %LINK-3-UPDOWN: Interface FastEthernet2/0, changed state to up
*Mar  8 06:55:14.335: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet2/0, changed state to up
R1(config-if)#int f2/1
R1(config-if)#no shut
R1(config-if)#
*Mar  8 06:55:26.007: %LINK-3-UPDOWN: Interface FastEthernet2/1, changed state to up
R1(config-if)#
*Mar  8 06:55:27.007: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet2/1, changed state to up
R1(config-if)#
```



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int f1/0
R1(config-if)#shut
R1(config-if)#
*Mar  8 06:52:41.203: %LINK-5-CHANGED: Interface FastEthernet1/0, changed state to administratively down
*Mar  8 06:52:42.203: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to down
R1(config-if)#
```



Network Intrusion Alert

ai.network.alert@gmail.com
Unauthorized interface ACTIVE -> FastEthernet2/0
12:58 (2 hours ago)

ai.network.alert@gmail.com
to me
12:58 (2 hours ago)

ai.network.alert@gmail.com
to me
15:06 (1 minute ago)
Trusted interface DOWN -> FastEthernet1/0
Trusted interface DOWN -> FastEthernet1/1
Unauthorized interface ACTIVE -> FastEthernet2/0
Unauthorized interface ACTIVE -> FastEthernet2/1

Fig 4. Email alert occured when the trusted and untrusted interface is down and up respectively

VI. EXISTING SYSTEM VS PROPOSED SYSTEM

Aspect	Existing System	Proposed System
Configuration	Manual CLI	Automated using intent
Ease of Use	Complex	User-friendly
Time	Slow	Fast
Errors	High	Reduced
Monitoring	Limited	Integrated
Security	Basic	Enhanced with alerts

Table 1. Aspects of existing system and proposed system

VII. PERFORMANCE EVALUATION

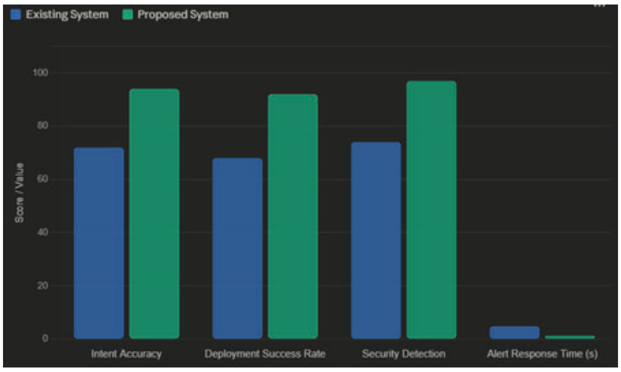


Fig 5. performance evaluation of existing system and proposed system

VIII. CONCLUSION

The proposed system provides an effective solution for simplifying network management by integrating intent-based automation with continuous monitoring. It enables administrators to give high-level instructions in natural language, which are automatically converted into device-specific configuration commands and deployed efficiently. This approach reduces manual effort, minimizes human errors, and improves the overall efficiency of network configuration.

In addition to automation, the system continuously monitors network interfaces to detect issues such as unauthorized activity and device failures. The alert mechanism ensures timely notification, allowing quick response to potential problems. By combining automation, monitoring, and alerting in a single framework, the system enhances network reliability, security, and ease of management in modern network environments.

IX. REFERENCES

[1] E. J. Scheid et al., “INSpIRE: Integrated NFV-based intent refinement environment, ” in Proc. IFIP/IEEE Symp. Integr. Netw. Service Manag., 2017.

[2] M. Cheminod, L. Durante, L. Seno, F. Valenza, and A. Valenzano, “A comprehensive approach to the automatic refinement and verification of access control policies, ” Comput. Security, vol. 80, pp. 186–199, Jan. 2019.

[3] J. Kim et al., “IBCS: Intent-based cloud services for security applications, ” IEEE Commun. Mag., vol. 58, no. 4, pp. 45–51, Apr. 2020.

[4] A. S. Jacobs et al., “Hey, Lumi! Using natural language for intent based network management, ” in Proc. USENIX ATC, Jul. 2021.

[5] A. Leivadeas and M. Falkner, “A survey on intent based networking, ” IEEE Commun. Surveys Tuts., vol. 25, no. 1, pp. 625655, 1st Quart., 2023.

[6] K. Mehmood, K. Kralevska, and D. Palma, “Intent-driven autonomous network and service management in future cellular networks: A structured literature review, ” Comput. Netw., vol. 220, Jan. 2023.