Network Working Group
Internet-Draft
Intended status: Informational
Expires: 18 August 2026

G. Zeng
Huawei

February 2026

MCP for Network Management: Problem Statement, Use Cases, and Requirements
draft-zeng-nmrg-mcp-usecases-requirements-00

Abstract

The emergence of large language models (LLMs) and AI agents is reshaping how network operators interact with infrastructure. However, current network management systems lack a standardized, secure, and intent-driven interface that enables AI agents to discover, invoke, and reason over network capabilities. This document presents a problem statement for integrating the Model Context Protocol (MCP) into network management, outlines key use cases—including troubleshooting, measurement, security, and optimization—and specifies functional, security, and interoperability requirements for MCP-based network management architectures.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 5 August 2026.

Copyright Notice

Table of Contents

1. Introduction

Traditional network management relies on protocol-specific interfaces (e.g., SNMP, NETCONF, RESTCONF) and manual scripting, which are rigid, siloed, and ill-suited for natural-language-driven automation. The Model Context Protocol (MCP), originally designed to standardize tool interaction for AI agents, offers a promising abstraction layer to unify network capabilities as callable “tools” and readable “resources.” This document formalizes the motivation, scenarios, and technical prerequisites for adopting MCP in network management.

2. Problem Statement

Modern networks face increasing complexity due to scale, heterogeneity, and dynamic service demands. Operators struggle with:

* *Intent-to-Action Gap*: Natural-language requests (e.g., “Fix slow video calls in Building B”) cannot be directly translated into coordinated network operations across devices and domains.

* *Tool Fragmentation*: Capabilities like ping, route lookup, or ACL modification exist but are exposed via disparate protocols (CLI, YANG RPCs, gNMI), making them inaccessible to generic AI agents.

* *Lack of Contextual Awareness*: Current systems do not provide structured, machine-readable context (e.g., topology, policy constraints, historical baselines) needed for intelligent reasoning.

* *Multi-Vendor Interoperability Gap*: In heterogeneous networks, the lack of a common semantic and syntactic interface for management operations forces operators to develop and maintain vendor-specific automation scripts. This drastically reduces operational efficiency and blocks the deployment of unified AI-driven management across domains.

Existing management frameworks address parts of these issues but lack a lightweight, agent-centric protocol that decouples AI logic from network implementation details. MCP fills this gap by providing a uniform, JSON-RPCbased interface for AI agents to securely interact with network elements as first-class tools.

Without standardizing how networks expose capabilities to AI agents, the industry risks fragmented, vendor-specific integrations that hinder interoperability, auditability, and safe automation.

3. Use Cases

This section describes four representative use cases where MCP enables intelligent, intent-driven network management.

3.1. Troubleshooting

An operator reports: “Users in Floor 3 cannot reach the cloud application.”

Under this scenario, the MCP Workflow runs as follows: The LLM interprets the intent and identifies required tools: get_interface_status, ping, show_route, check_acl. The MCP Client (in controller or local device) invokes these tools across relevant switches and routers. In D2D (Device-to-Device) mode, affected devices collaboratively gather data even if the controller is unreachable. The LLM correlates results and concludes: “ACL ‘block_external’ on Switch-F3 denies outbound traffic.” A repair suggestion is generated and presented for approval.

The MCP Workflow reduces mean time to resolution (MTTR) from hours to minutes with explainable root-cause analysis.

3.2. Network Measurement

An operator requests: “Monitor end-to-end latency between all branch offices every 5 minutes and alert if >50ms.”

Under this scenario, the MCP Workflow runs as follows: The request is parsed into a recurring measurement task. MCP Tools such as measure_latency, collect_interface_stats, and query_bgp_rib are scheduled across edge routers. Structured results (with timestamps, paths, and metadata) are returned via MCP Resources. The LLM detects anomalies using historical baselines and triggers alerts or auto-remediation.

The MCP Workflow enables continuous, intent-defined observability without custom collectors or polling scripts.

3.3. Security Operations

An operator requests: “Detect and isolate any device exhibiting abnormal outbound traffic patterns.”

Under this scenario, the MCP Workflow runs as follows: The security AI agent subscribes to flow data via MCP Resource netflow_records. Upon anomaly detection (e.g., sudden spike to unknown IP), it invokes get_device_info, list_connected_hosts, and apply_quarantine_acl. All actions are logged with cryptographic signatures for compliance. The agent may also query external threat intelligence via an external MCP Server.

The MCP Workflow enables real-time, closed-loop security response.

3.4. Network Optimization

An operator requests: “Optimize WAN bandwidth utilization during business hours without violating SLAs.”

Under this scenario, the MCP Workflow runs as follows: The optimizer agent collects real-time metrics (bandwidth_utilization, application_qos_stats) via MCP Resources. It simulates policy changes (e.g., adjusting QoS profiles, shifting SD-WAN paths) using sandboxed MCP Tools. After validation, it applies the optimal configuration via update_qos_policy or modify_sdwan_rules. Post-change telemetry confirms SLA compliance.

The MCP Workflow enables continuous, data-driven optimization aligned with business intent, reducing manual tuning.

4. Requirements

To support the above use cases safely and scalably, MCP-based network management must satisfy the following requirements.

4.1. Functional Requirements

FR1 Capability Exposure Every network element MUST expose its management capabilities as MCP Tools (for actions) and Resources (for state), mapped from underlying protocols (e.g., YANG, CLI).

FR2 Intent Interpretability Tools and Resources MUST include human- and machine-readable metadata (e.g., descriptions, parameter schemas) to enable accurate LLM parsing.

FR3 Safety Destructive operations require explicit user consent.

FR4 Progress Feedback Long-running operations (e.g., packet capture) MUST support asynchronous progress notifications.

4.2. Security Requirements

SR1 Authentication & Authorization TBD

SR2 Least Privilege Tools MUST be scoped to minimal necessary permissions (e.g., “read-only” vs “configure”).

4.3. Interoperability & Operational Requirements

IR1 Protocol Compliance MCP implementations MUST conform to the latest MCP specification (e.g., JSON-RPC 2.0 over TLS).

IR2 YANG Integration Tools/Resources SHOULD be describable via standardized YANG modules (e.g., ietf-mcp-nm).

IR3 Backward Compatibility MCP Servers MUST coexist with existing management protocols (NETCONF, SNMP).

IR4 Error Handling Clear error codes and recovery guidance MUST be provided for common failure modes (e.g., “tool not supported,” “resource locked”).

5. Security Considerations

This document’s requirements (Section 4.2) are designed to mitigate risks inherent in AI-driven network control, including prompt injection, unauthorized configuration changes, and data leakage. Additional considerations include:

* Avoiding over-reliance on LLM determinism; critical operations should require human-in-the-loop confirmation.

* Ensuring MCP Server implementations undergo formal security review before deployment in production networks.

6. IANA Considerations

TBD

7. References

[MCP-SPEC] Anthropic, "Model Context Protocol Specification", June 2025, <https://modelcontextprotocol.io/specification>.

[I-D.yang-nmrg-mcp-nm] YUANYUANYANG, Wu, Q., Lopez, D., Moreno, N. R., Tailhardat, L., and G. Zeng, "Applicability of MCP for the Network Management", Work in Progress, Internet-Draft, draft-yang-nmrg-mcp-nm-01, October 2025, <https://datatracker.ietf.org/doc/html/draft-yang-nmrg-mcp-nm-01>.

[I-D.zm-rtgwg-mcp-network-measurement] Zeng, G., Mao, J., Liu, B., Geng, N., Shang, X., Gao, Q., and Z. Li, "MCP-based Network Measurement Framework: Using Model Context Protocol for Intelligent Network Measurement", Work in Progress, Internet-Draft, draft-zm-rtgwg-mcp-network-measurement-01, November 2025, <https://datatracker.ietf.org/doc/html/draft-zm-rtgwg-mcp-network-measurement-01>.

[I-D.zm-rtgwg-mcp-troubleshooting] Zeng, G., Mao, J., Liu, B., Geng, N., Shang, X., Gao, Q., and Z. Li, "Using the Model Context Protocol (MCP) for Intent-Based Network Troubleshooting Automation", Work in Progress, Internet-Draft, draft-zm-rtgwg-mcp-troubleshooting-01, November 2025, <https://datatracker.ietf.org/doc/html/draft-zm-rtgwg-mcp-troubleshooting-01>.

Author’s Address

Guanming Zeng
Huawei
Email: zengguanming@huawei.com