The Eastasouth Journal of Information System and Computer Science
Vol. 1, No. 02, December, pp. 260-267

From Observability to Closed-Loop AIOps: Data-Driven Automation for Secure and Resilient Network Operations

Mohit Bajpai
Independent researcher, Alpharetta, GA, USA

Article Info

Article history:

Received Dec, 2023
Revised Dec, 2023
Accepted Dec, 2023

Keywords:

AIOps, Machine Learning;
Closed-Loop Automation;
Grafana;
NetFlow;
Network Operations
Network Security
Observability
OpenTelemetry
Prometheus
Root Cause Analysis
Splunk
Streaming Telemetry
Telemetry;
Zero Trust

ABSTRACT

Modern enterprise and service-provider networks are now distributed across cloud, edge, software-defined data centers, mobile access, Internet of Things (IoT), and hybrid work environments. The operational challenge is no longer limited to device availability; teams must interpret high-volume telemetry, fast-changing application paths, user-experience signals, identity context, security events, and configuration drift at machine speed. This updated article expands the original discussion of AI Ops, machine learning, observability, and network security by adding a data-centered reference architecture, operational metrics, model-selection considerations, security controls, deployment phases, and governance requirements. The article explains how telemetry from SNMP, streaming telemetry, NetFlow/IPFIX, syslog, OpenTelemetry, endpoint logs, cloud logs, configuration repositories, and security tools can be converted into actionable intelligence through anomaly detection, forecasting, causal correlation, risk scoring, and policy-based automation. It also positions closed-loop AIOps as a practical operating model that improves mean time to detect, mean time to acknowledge, mean time to resolve, service-level compliance, capacity planning, and security response while preserving human approval for high-risk actions.

Corresponding Author:

Name: Mohit Bajpai
Institution: Independent researcher, Alpharetta, GA, USA
Email: bajpaimohit@gmail.com

1. INTRODUCTION

The original paper correctly identified that artificial intelligence, machine learning, and observability can transform network operations by moving teams away from static rules and reactive troubleshooting. This updated version strengthens that position by treating observability data, configuration state, topology, identity, and security context as a unified evidence base for automated decision-making. In current network environments, the same outage may involve a WAN circuit, an SD-WAN policy, an identity provider, DNS, firewall inspection, application latency, a cloud region, and an endpoint protection agent. A traditional network operations center may see these as separate alarms, while an AIOps platform should correlate them into a single operational narrative [1]–[5].

Cisco projected that by 2023 the world would reach 5.3 billion Internet users, 29.3 billion networked devices, and 3.6 devices and connections per capita. This growth illustrates why manual monitoring and rule-only alerting cannot scale with the volume and diversity of devices, protocols,

and user behaviors now present in enterprise networks. Security pressure has increased at the same time: the IBM Cost of a Data Breach Report 2023 placed the global average cost of a breach at USD 4.45 million, and ENISA identified DDoS and ransomware among the highest-ranked prime threats during its 2023 reporting period. These data points show that automation is not simply an efficiency improvement; it is a resilience requirement [6].

AIOps should not be interpreted as replacing network engineers. Instead, it augments engineers by reducing noise, prioritizing risk, generating probable causes, recommending remediation, and executing safe changes where policy allows. The most effective model is a progressive closed-loop approach: observe, understand, decide, act, verify, and learn [2], [5], [7].

2. NETWORK OPERATIONS AND SECURITY CHALLENGES

Network teams face several structural challenges that make conventional monitoring insufficient [6], [8]–[12].

a. Scale and telemetry volume: Metrics, logs, traces, packet metadata, flow records, configuration states, and security events grow faster than human analysts can inspect [6], [13], [14].
b. Dynamic traffic patterns: SaaS adoption, remote access, content delivery networks, and cloud workloads make baselines highly contextual by time, location, application, and user group [6], [11], [12].
c. Distributed ownership: Network, cloud, security, identity, application, and endpoint teams often use separate tools, which increases handoff delay during incidents [12]–[15].
d. Alert fatigue: Static thresholds create large volumes of duplicate or low-value alerts, especially during upstream failures where many dependent systems alarm at the same time [5], [16], [17].
e. Security convergence: Availability incidents and security incidents increasingly overlap, particularly in cases involving DDoS, credential abuse, lateral movement, misconfiguration, and exfiltration [9], [10], [18].
f. Configuration risk: Many outages originate from technically valid but operationally harmful changes, such as an incorrect route policy, firewall rule, DNS change, QoS class, or SD-WAN steering policy [2], [5], [19].

3. DATA FOUNDATION FOR AIOPS AND OBSERVABILITY

AIOps quality depends directly on the quality and diversity of data. A practical implementation should ingest both performance and security signals, then normalize them into a common entity model. Device identifiers, interface names, timestamps, topology references, software versions, site names, application tags, and business service labels must be standardized before machine learning models are trusted [1], [5], [13], [14].

3.1 Core Telemetry Sources

Table 1. Classification of Telemetry Sources for Operational Monitoring and Security Analysis

Telemetry source	Typical examples	Operational value	Security value
Metrics	SNMP counters, Prometheus metrics, streaming telemetry, interface errors, CPU, memory, buffer drops	Capacity planning, SLO monitoring, congestion detection	Detection of abnormal traffic spikes or resource exhaustion

Telemetry source	Typical examples	Operational value	Security value
Logs	Syslog, device logs, firewall logs, authentication logs, DNS logs, cloud logs	Fault sequence reconstruction and change impact analysis	Credential abuse, policy violations, malware callbacks
Traces	OpenTelemetry traces, application path traces, service dependency traces	Application-to-network latency correlation	Suspicious service-to-service access patterns
Flows	NetFlow, IPFIX, sFlow, VPC flow logs	Traffic engineering, path visibility, application usage	DDoS, scanning, exfiltration, lateral movement
Configuration state	Git repositories, device snapshots, CMDB, golden templates	Drift detection, compliance, rollback decisioning	Unauthorized change detection and segmentation validation
Topology and dependency	LLDP/CDP, routing peers, SD-WAN overlay, Kubernetes services, cloud VPC maps	Blast-radius calculation and probable-cause ranking	Attack-path analysis and control placement

3.2 Data Quality Controls

Before applying ML, the platform should validate telemetry completeness, timestamp consistency, source authenticity, schema drift, duplicate events, and outlier frequency. Incomplete or stale data can cause false positives, while mislabeled entities can lead to incorrect remediation. Recommended controls include schema validation, time synchronization checks, missing-metric detection, source reputation scoring, configuration versioning, and sampling-rate visibility [5], [13], [14].

4. AI/ML TECHNIQUES FOR NETWORK OPERATIONS

No single model solves every network operations problem. Effective AIOps platforms combine statistical methods, classical machine learning, time-series models, graph analytics, and rules-based policy controls. Models should be selected according to the operational question and the available labels [1], [5], [16]–[18], [20].

Table 2. Mapping of AI Techniques to Network Operations and Security Use Cases

Use case	Preferred techniques	Required data	Output	Example action
Anomaly detection	Isolation Forest, autoencoders, robust z-score, seasonal decomposition	Metrics, logs, flows, historical baselines	Anomaly score and affected entities	Open incident, enrich alert, request validation
Traffic forecasting	ARIMA, Prophet-style decomposition, LSTM, gradient boosting	Time-series traffic, calendar, business events	Predicted utilization and confidence interval	Pre-scale capacity or adjust SD-WAN path
Root cause analysis	Bayesian networks, graph ranking, causal inference, event correlation	Topology, dependencies, alerts, config changes	Ranked probable causes	Route to owner, attach evidence, propose rollback
Security risk scoring	Supervised classifiers, behavior analytics, threat-intel enrichment	Auth logs, EDR, firewall logs, flows, vulnerability data	Risk score by user/device/session	Block, challenge MFA, isolate, rate-limit

Use case	Preferred techniques	Required data	Output	Example action
Change-risk prediction	Classification, similarity search, historical change outcomes	Change records, config diffs, incident history	Risk category and reviewer suggestions	Require approval, run pre-checks, stage rollout
Automated remediation	Policy engine, runbooks, reinforcement learning only in constrained domains	Validated findings, safe actions, runbook history	Action plan and expected impact	Execute low-risk runbook or request approval

5. OBSERVABILITY INTEGRATION ARCHITECTURE

The updated architecture extends the original deployment diagram by separating the data plane, observability plane, intelligence plane, automation plane, governance plane, and feedback plane. This separation is important because the platform must support both real-time detection and controlled remediation. In production, the automation plane should never act directly on raw model output; it should act on validated decisions that meet policy, confidence, and blast-radius constraints [3], [7], [13], [14].

Table 3. Updated reference architecture for observability-driven closed-loop AIOps in network operations and security

1. Network / Cloud / Security Sources	2. Telemetry Collection and Normalization	3. Observability and Storage
Routers, switches, firewalls, SD-WAN, cloud VPCs, Kubernetes, endpoints, identity systems	SNMP, streaming telemetry, syslog, NetFlow/IPFIX, OpenTelemetry, config snapshots, CMDB enrichment	Prometheus, Splunk, Elastic, Grafana, data lake, topology graph, time-series store
4. AIOps Intelligence Plane	**5. Decision and Policy Plane**	**6. Automation and Orchestration**
Anomaly detection, forecasting, correlation, RCA, risk scoring, change-risk analysis	Confidence thresholds, approval gates, zero-trust controls, blast-radius rules, audit logging	Ansible, Terraform, Kubernetes, SDN/SD-WAN APIs, SOAR, ticketing and chat workflows
7. Verification Layer	**8. Continuous Learning**	**9. Governance**
Post-action telemetry checks, SLO validation, rollback triggers	Feedback labels, model retraining, runbook tuning, false-positive reduction	Model registry, access control, compliance evidence, human review, audit trails

6. SECURITY INTEGRATION: FROM NETWORK MONITORING TO ADAPTIVE DEFENSE

Network security benefits from AIOps when telemetry is linked to identity, asset criticality, vulnerability context, and business service impact. NIST Zero Trust Architecture emphasizes that trust should not be granted implicitly based only on network location. This aligns with AIOps because decisions should be based on continuously evaluated signals rather than static perimeter assumptions [9], [10], [18], [21].

In a security-aware AIOps implementation, a flow anomaly is not enough by itself to trigger a high-risk action. The platform should evaluate whether the endpoint is managed, whether the user recently failed authentication, whether the destination is unusual, whether the asset has known vulnerabilities, and whether a similar pattern has appeared elsewhere. This contextual scoring reduces false positives and supports automated containment actions such as micro-segmentation, rate limiting, firewall rule adjustment, user session revocation, or ticket escalation [9], [10], [18], [21].

7. OPERATIONAL METRICS AND DATA-DRIVEN VALUE

The business case for AIOps should be measured through operational outcomes instead of only model accuracy. Model accuracy is useful, but network leaders need to know whether the system reduced downtime, accelerated troubleshooting, reduced alert noise, improved change safety, and strengthened security response.

Table 4. AIOps Performance Metrics and Operational Improvement Indicators

Metric	Definition	Why it matters	How AIOps improves it
MTTD	Mean time to detect an incident	Measures monitoring speed and visibility	Real-time anomaly detection and correlation
MTTA	Mean time to acknowledge	Measures alert triage efficiency	Noise reduction and severity prioritization
MTTR	Mean time to resolve	Measures restoration speed	RCA ranking, runbook recommendations, closed-loop remediation
Alert compression ratio	Raw alerts divided by correlated incidents	Measures alert fatigue reduction	Deduplication, dependency mapping, temporal grouping
Change failure rate	Percentage of changes causing incidents or rollback	Measures change quality	Risk scoring and pre-change simulation
SLO compliance	Percentage of time service objectives are met	Measures customer experience	Forecasting, capacity management, proactive action
False positive rate	Incorrect alerts divided by total alerts	Measures model and rule usefulness	Feedback labels and continuous model tuning

8. EXAMPLE END-TO-END SCENARIO

Consider an enterprise using SD-WAN, cloud-hosted applications, and identity-based access. Users in one region report intermittent application latency. Traditional tools may produce separate alerts for packet loss, firewall session growth, VPN tunnel instability, DNS latency, and application timeout. A closed-loop AIOps workflow can process the event as follows [2], [3], [5], [7], [13], [14]:

1. Telemetry ingestion detects increased packet loss on two WAN underlay circuits, elevated retransmissions, and a simultaneous configuration change on an SD-WAN policy.
2. The correlation engine suppresses duplicate interface alarms and links the symptoms to one business service and one geographic region
3. The root-cause model ranks the SD-WAN policy change as the most likely cause because the anomaly begins within minutes of deployment and affects traffic steered through the updated rule.
4. The decision plane checks blast radius, recent change history, model confidence, and rollback safety.
5. For a low-risk rollback, the automation plane restores the previous policy. For a high-risk production service, the system requests engineer approval and attaches evidence.
6. The verification layer confirms that latency and retransmissions return to baseline, then updates the incident record and labels the model outcome for future learning.

9. DEPLOYMENT ROADMAP

A successful deployment should be phased. Organizations that attempt full autonomy before establishing data quality, governance, and runbook maturity usually

create operational risk. A recommended roadmap is shown below [5], [7], [19].

Table 5. Phased Implementation Roadmap for AIOps Adoption

Phase	Objective	Key activities	Exit criteria
Phase 1: Visibility	Create trustworthy telemetry baseline	Inventory data sources, normalize entities, build dashboards, define SLOs	Critical services visible with reliable telemetry
Phase 2: Correlation	Reduce alert noise	Map topology, dependency graph, event grouping, alert deduplication	Correlated incidents replace raw alarm floods
Phase 3: Prediction	Forecast risk and capacity	Train models, evaluate drift, validate thresholds, build confidence intervals	Predictions are accurate enough for human-assisted decisions
Phase 4: Assisted remediation	Recommend actions	Integrate runbooks, approvals, ticketing, chatops, evidence bundles	Engineers accept recommendations and feedback is captured
Phase 5: Closed-loop automation	Automate safe actions	Policy gates, rollback, audit logging, post-action verification	Low-risk actions execute automatically with measured improvement

10. MODEL GOVERNANCE, RISK, AND LIMITATIONS

AIOps introduces new governance requirements. Models can drift when traffic patterns, software releases, user behavior, or routing policies change. Models can also inherit bias from historical incident labels or fail when telemetry sources disappear. Therefore, production AIOps should include model versioning, approval workflows, drift detection, confidence scoring, and rollback capability.

a. Human-in-the-loop control should remain mandatory for high-impact changes such as route redistribution, firewall policy modification, identity enforcement, and production-wide rollback.
b. Model decisions should be explainable enough for engineers to verify why an incident was grouped, why a cause was ranked highly, and why a remediation was recommended.
c. Security teams should validate those automated actions cannot be abused by attackers who intentionally generate telemetry patterns to trigger harmful runbooks.
d. Data retention and privacy requirements must be addressed because observability data may contain IP addresses, usernames, device identifiers, or application metadata.

11. UPDATED TECHNICAL RECOMMENDATIONS

a. Build a unified telemetry model before training advanced models. Entity normalization is more important than algorithm complexity in early deployments.
b. Use topology-aware correlation. Network incidents are dependency problems, so graph context is essential for root-cause ranking.
c. Separate detection from action. Anomaly scores should feed a policy and decision layer before remediation occurs.
d. Start with low-risk automation such as ticket enrichment, duplicate suppression, evidence collection, and dashboard links before enabling configuration changes.
e. Measure value with operational metrics such as MTTR, alert

compression, SLO compliance, and change failure rate.

f. Apply zero-trust principles to network automation. Every remediation action should be authenticated, authorized, scoped, logged, and verified.
g. Create feedback loops. Engineers should be able to label incidents, mark false positives, reject recommendations, and feed that data back into model improvement.

12. CONCLUSION

AI Ops, machine learning, and observability provide a practical path toward more autonomous, secure, and resilient network operations. The most important change is not simply adding algorithms to monitoring tools; it is building a disciplined operating model that connects telemetry, topology, configuration, security context, business impact, policy, automation, and continuous learning. When implemented with governance and phased adoption, AIOps can reduce alert fatigue, accelerate root-cause analysis, improve capacity planning, strengthen security response, and support closed-loop remediation for well-understood and low-risk conditions. As networks continue to expand across cloud, edge, IoT, hybrid work, and software-defined infrastructure, observability-driven automation will become a core capability for maintaining service reliability and security at scale.

REFERENCES

[1] K. M. Sivalingam, "Applications of Artificial Intelligence, Machine Learning and related techniques for Computer Networking Systems," *arXiv*. 2021. doi: 10.48550/arXiv.2105.15103.

[2] N. Feamster and J. Rexford, "Why (and how) networks should run themselves," 2018. doi: 10.1145/3232755.3234555.

[3] D. Rossi and L. Zhang, "Landing AI on networks: An equipment vendor viewpoint on autonomous driving networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 19, no. 3, pp. 3670–3684, 2022, doi: 10.1109/TNSM.2022.3169988.

[4] G. Luo, Q. Yuan, J. Li, S. Wang, and F. Yang, "Artificial Intelligence Powered Mobile Networks: From Cognition to Decision," *IEEE Netw.*, vol. 36, no. 3, pp. 136–144, 2022, doi: 10.1109/MNET.013.2100087.

[5] P. Notaro, J. Cardoso, and M. Gerndt, "A survey of AIOps methods for failure management," *ACM Trans. Intell. Syst. Technol.*, vol. 12, no. 6, pp. 1–45, 2021.

[6] Cisco, "Cisco Annual Internet Report (2018-2023) White Paper," Cisco Systems, 2020.

[7] ETSI, "Zero-touch network and Service Management (ZSM); Reference Architecture (ETSI GS ZSM 002)," European Telecommunications Standards Institute, 2019.

[8] I. B. M. Security, "Cost of a Data Breach Report 2023," IBM Corporation, 2023.

[9] Verizon, "2023 Data Breach Investigations Report," Verizon Business, 2023.

[10] E. U. A. for Cybersecurity, "ENISA Threat Landscape 2023," ENISA, 2023.

[11] D. Kreutz, F. M. V Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, 2014.

[12] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 236–262, 2016, doi: 10.1109/COMST.2015.2477041.

[13] P. Authors, "Prometheus documentation: Data model and monitoring concepts," Cloud Native Computing Foundation, 2023.

[14] C. N. C. Foundation, "OpenTelemetry documentation and project updates: Metrics, logs, and traces for cloud-native observability," CNCF, 2023.

[15] D. Kreutz, F. M. V Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, 2015, doi: 10.1109/JPROC.2014.2371999.

[16] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009, doi: 10.1145/1541880.1541882.

[17] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Comput. Networks*, vol. 51, no. 12, pp. 3448–3470, 2007, doi: 10.1016/j.comnet.2007.02.001.

[18] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016, doi: 10.1109/COMST.2015.2494502.

[19] M. Bajpai, "Network Infrastructure and Disaster Recovery Planning for Seasonal Events," *Eur. J. Adv. Eng. Technol.*, vol. 8, no. 11, pp. 132–136, 2021.

[20] M. Aledhari, R. Razzak, and R. M. Parizi, "Machine learning for network application security: Empirical evaluation

and optimization," *Comput. Electr. Eng.*, vol. 91, p. 107052, 2021, doi: 10.1016/j.compeleceng.2021.107052.

[21] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture (NIST Special Publication 800-207)," National Institute of Standards and Technology, 2020. doi: 10.6028/NIST.SP.800-207.