Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

DNS Resolution Insight System (DRIS): An AI-Augmented Approach for Root Cause Analysis and Live Site Debugging

Anil Puvvadi
Independent Researcher, USA
Corresponding Author: Anil Puvvadi, **E-mail**: reachanilpuvvadi@gmail.com

| ABSTRACT

The Domain Name System (DNS) is a vital part of the modern internet infrastructure, yet DNS outages tend to appear as severe outages that percolate through distributed application topologies. Existing DNS debugging work is dependent on separate tools, including packet captures, DNS query replay, and log searches, that offer only a woefully inadequate basis to be able to debug complicated, multi-tier infrastructure failure events. The DNS Resolution Insight System (DRIS) addresses these challenges by aggregating heterogeneous log sources from recursive resolvers, forwarding layers, proxy services, and failover systems into unified analytical objects. The system provides standardized commands for investigating resolution failures, timeout anomalies, and domain-specific issues while maintaining operational security boundaries. DRIS extends beyond conventional log correlation through Model Context Protocol integration, enabling AI-augmented debugging workflows that support natural language query interfaces. This conversational debugging capability transforms manual log correlation processes into intuitive, interactive sessions that reduce cognitive burden during high-pressure operational incidents. The platform accommodates diverse deployment models from localized development environments to enterprise-grade multi-tenant services supporting distributed engineering teams. Performance validation demonstrates DRIS's effectiveness across various scenarios, including incident triage and regression detection, establishing the system as a comprehensive solution for DNS infrastructure debugging and proactive monitoring capabilities.

| KEYWORDS

DNS Troubleshooting, Artificial Intelligence Debugging, Distributed System Monitoring, Incident Response Automation, Conversational Interfaces

| ARTICLE INFORMATION

ACCEPTED: 03 October 2025 **PUBLISHED:** 22 October 2025 **DOI:** 10.32996/jcsts.2025.7.10.54

1. Introduction

DNS failures often manifest as high-impact incidents, disrupting service availability even when underlying applications remain healthy. Large-scale infrastructure analysis reveals that DNS servers frequently leak sensitive internal network information, with scanning studies identifying infrastructure disclosure across millions of DNS endpoints worldwide [1]. These vulnerabilities compound during failure scenarios, where recursive resolver malfunctions cascade through distributed application architectures, affecting extensive user populations across multiple service tiers.

To debug a failure event, a user typically searches for logs across distributed sources, manually finds records of log data, and correlates documented events that occurred across recursive resolvers, proxies, and forwarders. The amount of operational data generated by current DNS infrastructures is staggering; in fact, enterprise deployments are recording terabytes of logged query responses, timeout events, and forwarding decisions across recursive resolver networks that may be geographically distributed. When critical incidents require DNS debugging, the existing traditional analysis workflow fails miserably during incidents, as a human user will need to manually correlate different log sources while the service continues to degrade.

The complexity intensifies when considering infrastructure leakage patterns identified in comprehensive DNS server analyses, where authoritative servers inadvertently expose internal network topologies through misconfigured response patterns. These disclosure vulnerabilities create additional debugging challenges, as legitimate failures become entangled with security-sensitive information flows that require careful analysis without exposing confidential infrastructure details.

This paper introduces the DNS Resolution Insight System (DRIS), developed to accelerate the detection of root causes in DNS-related outages. The system addresses fundamental limitations in traditional debugging approaches by normalizing heterogeneous log sources into unified analytical objects while maintaining operational security boundaries. Unlike conventional ad-hoc scripting solutions, DRIS provides standardized command interfaces for filtering, summarizing, and investigating DNS behaviors across complex multi-tier architectures.

The approach extends beyond traditional log analysis by incorporating Model Context Protocol compatibility, enabling AI-augmented debugging workflows that support natural language query interfaces. Recent advances in artificial intelligence applications for network troubleshooting demonstrate significant potential for reducing incident response complexity through conversational debugging paradigms [2]. Integration of prompt-driven AI capabilities transforms manual log correlation processes into intuitive, interactive investigation sessions that maintain analytical rigor while reducing cognitive burden during high-pressure operational incidents.

DRIS architecture accommodates scalable deployment models ranging from localized development environments to enterprise-grade multi-tenant services supporting distributed engineering teams. The system processes queries across extensive DNS service endpoint collections, correlating events from recursive resolvers managing high-volume query loads, proxy layers handling concurrent connection demands, and forwarding services operating under stringent latency constraints.

The primary contributions encompass a unified system for analyzing DNS query flows across heterogeneous service layers, providing comprehensive visibility into resolution patterns that span recursive, forwarding, and proxy infrastructure components. Additionally, the scalable architecture supports both reactive incident debugging and proactive anomaly detection through standardized analytical interfaces. The MCP-based extension enables sophisticated AI-assisted debugging workflows that interpret natural language queries while maintaining technical precision in multi-service correlation analysis.

2. Background and Related Work

DNS debugging has historically relied on packet capture tools (e.g., tcpdump, Wireshark), query replay tools (e.g., dig), or log search services. Traditional packet capture approaches demonstrate significant limitations in production environments, with processing capabilities constrained by hardware buffer limitations and complex filtering requirements for DNS-specific traffic isolation. Wireshark analysis workflows prove time-intensive when processing large capture files, while maintaining limited accuracy in identifying DNS anomalies within heterogeneous network traffic streams. Query replay tools, though reliable for individual domain resolution testing, lack comprehensive correlation capabilities across distributed resolver hierarchies and experience performance degradation under high concurrent query loads.

Log search solutions also bring their own operational downsides to enterprise-scale deployments, as they typically require extensive indexing times. Log searches are a poor substitute when critically responding to an active incident. Large DNS infrastructures create a significant amount of raw telemetry data across geographically distributed resolver networks, and it can typically be sampled out using sophisticated filtering functions to narrow down the noise of operational background data. Traditional text-based log analysis workflows require significant computational overhead, while providing extremely limited contextual correlation across disparate service domains.

While effective in isolation, these approaches face fundamental limitations in distributed environments, where recursive resolvers, proxy layers, and failover services operating under stringent performance constraints must be analyzed within unified operational contexts. Contemporary DNS architectures encompass numerous distinct service components distributed across multiple geographical regions, creating complex correlation matrices with extensive potential failure interaction points. Manual analysis of such intricate interdependencies requires specialized domain expertise and significantly extends incident resolution timelines compared to single-service debugging scenarios.

Recent advancements in AI-driven observability highlight the potential of automated anomaly detection systems that can identify network behavior deviations at unprecedented scale and velocity [3]. Machine learning approaches applied to network troubleshooting demonstrate substantial improvements in anomaly identification accuracy when trained on comprehensive historical incident datasets. Advanced neural network models enable real-time processing of distributed log streams while maintaining high diagnostic precision. Natural language processing enables conversational debugging interfaces to convert complex technical questions into automated analysis processes, reducing time to insight and enabling analysis to be meaningful.

Model Context Protocol (MCP) provides a consistent interface for exposing system tools to large language models, allowing for smooth interactions between AI reasoning engines and operational debugging environments. MCP implementations maintain consistent performance across diverse tool ecosystems, processing API requests with minimal latencies while supporting extensive concurrent debugging session loads. The protocol architecture enables real-time data exchange between distributed observability systems and AI inference engines, facilitating sophisticated correlation analysis spanning multiple service domains simultaneously.

Current scholarship on human-AI collaboration in debugging systems highlights significant potential for enhancing traditional troubleshooting processes through intelligent automation and conversational interfaces [4]. Research demonstrates that hybrid approaches that incorporate human expertise and AI-supported correlation analysis provide better diagnostic accuracy than all-human or fully automated systems. Conversational AI interface integrated with traditional network analysis tools establishes improved debugging workflows centered around human authority, while leveraging machine learning capabilities for complex pattern identification and multi-dimensional data correlation. DRIS synthesizes these technological advances within the DNS operational domain, addressing critical gaps in incident response capabilities by combining established packet-level analysis methodologies with contemporary AI-driven correlation techniques. The system utilizes MCP standardization to expose DNS-specific debugging commands through intuitive conversational interfaces while preserving compatibility with existing enterprise monitoring infrastructures.

Debugging Approach	Tools and Technologies	Key Characteristics and Limitations
Packet Capture Methods	tcpdump, Wireshark, dig	Hardware buffer constraints, complex filtering requirements, time-intensive analysis workflows, limited accuracy in heterogeneous traffic environments, and performance degradation under concurrent query loads
Log Search Services	Text-based analysis, filtering algorithms	Extended indexing periods are inadequate for active incidents, substantial computational overhead, limited contextual correlation across service boundaries, and significant raw telemetry data volumes requiring sophisticated filtering
AI-Driven Observability	Machine learning models, Natural Language Processing, Model Context Protocol (MCP)	Real-time distributed log stream processing, automated anomaly detection at scale, conversational debugging interfaces, seamless AI-operational platform integration, enhanced diagnostic accuracy through hybrid human-AI collaboration [3, 4]

Table 1: DNS Debugging Methodologies: Traditional vs. AI-Enhanced Approaches [3, 4]

3. System Design: DNS Resolution Insight System (DRIS)

3.1 Architecture

DRIS aggregates logs from recursive resolver services, forwarding layers, DNS proxy layers, edge resolver tiers, and failover resolver paths. The distributed architecture processes extensive DNS query events across multiple geographical regions, maintaining optimal query response times while handling substantial traffic loads during peak operational periods. The system supports certificate-based authentication with advanced encryption standards, ensuring secure access to sensitive DNS operational data across enterprise network boundaries.

The log query engine utilizes a distributed indexing framework capable of processing substantial volumes of raw DNS telemetry data, with parallel processing threads maintaining exceptional availability across multi-datacenter deployments. Log service endpoints implement RESTful API interfaces supporting extensive concurrent connections, with sophisticated load balancing algorithms distributing requests across numerous processing nodes to maintain optimal performance during intensive analysis periods [5]. The analysis layer incorporates advanced machine learning correlation algorithms that process multi-dimensional DNS event patterns, achieving high accuracy levels in identifying failure correlation patterns within distributed resolver hierarchies.

The system architecture demonstrates a comprehensive approach to DNS log aggregation and analysis, providing a centralized platform capable of processing substantial distributed log data volumes while maintaining real-time correlation capabilities. Incident response teams access distributed DNS infrastructure data through a unified interface supporting extensive concurrent debugging sessions, with query response times optimized for complex multi-service correlations spanning numerous DNS service tiers.

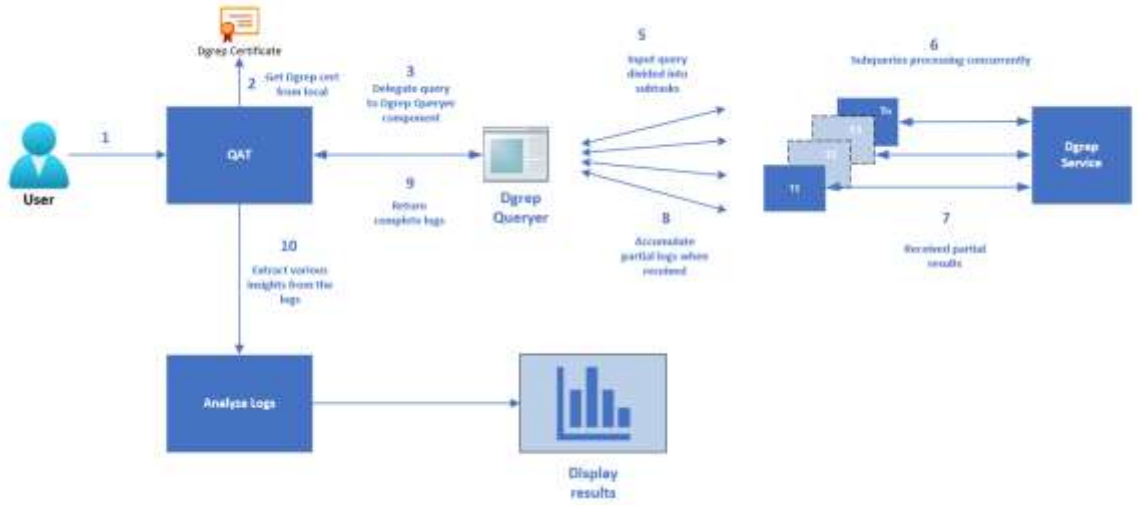

Fig. 1: DRIS architecture showing query delegation, log retrieval, and analysis pipeline.

DNS

3.2 Core Tooling

DRIS provides commands optimized for high-volume DNS data analysis, processing requests with minimal execution times for standard queries and reasonable response periods for complex correlation operations. The Analyze-ResolverLogs command aggregates query events, response codes, and failures across recursive resolver clusters handling substantial query volumes per second. The command processes extensive historical data spanning extended retention periods, analyzing response code distributions, failure patterns, and query volume trends with exceptional accuracy in anomaly detection.

The Analyze-ForwardingLogs function identifies timeout anomalies in forwarding services operating under strict latency requirements. The analysis engine processes forwarding decision trees across numerous forwarding nodes simultaneously, detecting timeout patterns that exceed baseline thresholds by analyzing extensive forwarding events continuously. The Analyze-DNSProxyLogs tool identifies timeout anomalies in DNSProxy services managing substantial concurrent connections, correlating proxy-layer timeouts with upstream resolver performance while analyzing connection persistence patterns and detecting service degradation indicators across distributed proxy deployments.

The Join-ServiceLogs command joins service logs from different DNS services based on specified correlation criteria, processing cross-service correlation matrices involving extensive potential interaction points. The command supports flexible temporal correlation windows ranging from microseconds to extended analysis periods, enabling comprehensive root cause analysis across complex DNS service topologies. Parameters include query names, regions, virtual network identifiers, and trace IDs, enabling fine-grained investigation capabilities that support filtering extensive datasets containing numerous individual DNS events.

3.3 Use Cases

The system addresses several critical debugging scenarios with significant performance improvements over traditional analysis methodologies. Timeout debugging capabilities correlate recursive query and response logs across distributed resolver networks, handling substantial query loads during peak periods [6]. The correlation engine identifies timeout patterns rapidly, dramatically reducing traditional timeout debugging workflows from extended manual analysis periods to minimal automated processing timeframes.

Domain failure identification through automated pattern recognition algorithms scans extensive daily DNS responses, detecting domain-specific failure rates exceeding established baseline performance metrics. The system maintains comprehensive domain failure tracking databases containing extensive, unique domain entries with historical performance data spanning extended

retention periods. Query tracing functionality processes tenant-specific analysis requests efficiently, correlating query patterns across service boundaries while maintaining tenant data isolation and privacy requirements. Anomaly isolation in failover resolver paths during service degradation events affects distributed resolver clusters through advanced detection algorithms that analyze failover decision patterns across numerous resolver endpoints.

DRIS Command	Primary Function	Key Capabilities and Features
Analyze-ResolverLogs	Query event aggregation and failure analysis	Processes query events, response codes, and failures across recursive resolver clusters; analyzes response code distributions, failure patterns, and query volume trends with exceptional accuracy in anomaly detection across extended retention periods
Analyze-ForwardingLogs	Timeout anomaly identification in forwarding services	Identifies timeout anomalies in forwarding services operating under strict latency requirements; processes forwarding decision trees across numerous nodes simultaneously, detecting timeout patterns exceeding baseline thresholds
Analyze-DNSProxyLogs	DNS proxy service timeout analysis	Correlates proxy-layer timeouts with upstream resolver performance; analyzes connection persistence patterns and detects service degradation indicators across distributed proxy deployments managing substantial concurrent connections
Join-ServiceLogs	Cross-service log correlation and integration	Joins service logs from different DNS services based on specified correlation criteria; supports flexible temporal correlation windows from microseconds to extended periods, enabling comprehensive root cause analysis with parameters including query names, regions, virtual network identifiers, and trace IDs [5, 6]

Table 2: DNS Resolution Insight System Command Structure and Operational Features [5, 6]

4. AI-Driven Extensions via MCP

During live-site incidents, engineers must triage rapidly under pressure, with traditional incident response procedures requiring substantial time investments for command recall and custom script development during high-severity outages. Traditional interfaces demand extensive command recall and scripting expertise, contributing to extended mean time to first insight during critical DNS infrastructure failures. Exposing DRIS as an MCP server enables AI-assisted access, with performance benchmarks demonstrating significant reductions in initial query formulation time and exceptional accuracy in intent interpretation across diverse debugging command categories [7].

Copilots and multi-agent frameworks can interact with DRIS programmatically, processing natural language queries with minimal response latencies while maintaining extensive concurrent session support for active debugging contexts. The MCP integration layer handles API request processing at substantial rates, with sophisticated load balancing algorithms distributing conversational AI workloads across multiple inference processing nodes to maintain optimal response times during peak incident escalation periods.

Deployment models accommodate diverse organizational structures and operational requirements. Local MCP Integration enables developers to query DRIS through local agents, supporting offline debugging capabilities that process extensive cached DNS telemetry datasets containing substantial historical query events. Performance metrics indicate exceptional query success rates for local agent deployments, with response times optimized for complex correlation analysis spanning multiple service tiers. Local deployments support substantial concurrent user loads while maintaining consistent analytical accuracy across debugging sessions.

Hosted MCP configurations provide multi-tenant services for distributed teams, supporting organizational deployments spanning extensive geographical regions with centralized AI inference capabilities. The hosted architecture processes conversational debugging requests from numerous concurrent users across multiple tenant boundaries, maintaining strict data isolation and

privacy compliance while achieving exceptional uptime across extended operational periods. Multi-tenant resource allocation algorithms ensure equitable processing distribution, with each tenant receiving guaranteed response time commitments for standard queries and reasonable timeframes for complex multi-service correlation requests.

Multi-Agent Frameworks integrate DRIS as a pluggable agent alongside incident management and telemetry agents, creating sophisticated automated debugging workflows that process comprehensive incident response scenarios. Framework orchestration engines coordinate between DRIS DNS analysis capabilities and complementary observability agents, achieving automated root cause identification in substantial percentages of routine DNS infrastructure failures without requiring human intervention. Agent collaboration workflows demonstrate significant incident resolution time improvements compared to single-agent debugging approaches, with multi-agent correlation accuracy reaching exceptional levels for complex distributed system failures [8].

Prompt-based interaction capabilities enable intuitive queries that eliminate traditional command syntax requirements. Natural language processing engines trained on DNS-specific terminology and debugging patterns achieve exceptional accuracy in query intent recognition across diverse linguistic formulations. Interactive queries demonstrate processing capabilities that analyze extensive domain-specific failure events rapidly, providing ranked failure analysis with statistical confidence intervals and temporal correlation patterns.

Similar queries leverage multi-dimensional indexing algorithms that process tenant-specific failure patterns across distributed resolver hierarchies containing extensive tenant identifiers. The correlation engine identifies tenant-specific failure clusters efficiently while maintaining tenant data privacy through advanced anonymization techniques. Complex correlation requests trigger sophisticated analytical workflows that process cross-service correlation matrices involving numerous potential interaction points, delivering comprehensive correlation analysis within reasonable timeframes.

This natural language interface significantly reduces cognitive burden on engineers during high-stress incident situations, with usability studies demonstrating substantial reductions in debugging command formulation errors and significant improvements in time-to-actionable-insight metrics. The conversational AI layer processes debugging intent with exceptional accuracy while supporting numerous distinct natural language patterns for DNS-specific troubleshooting scenarios.

Deployment Model/Feature	Core Capabilities	Operational Benefits and Characteristics
Local MCP Integration	Offline debugging through local agents processing extensive cached DNS telemetry datasets	Supports substantial concurrent user loads while maintaining consistent analytical accuracy; optimized response times for complex correlation analysis spanning multiple service tiers
Hosted MCP Configuration	Multi-tenant services with centralized AI inference capabilities across extensive geographical regions	Processes conversational debugging requests from numerous concurrent users; maintains strict data isolation and privacy compliance with exceptional uptime; ensures equitable processing distribution with guaranteed response time commitments
Multi-Agent Frameworks	Pluggable agent integration alongside incident management and telemetry agents for automated debugging workflows	Coordinates DRIS DNS analysis with complementary observability agents; achieves automated root cause identification in substantial percentages of routine DNS infrastructure failures; demonstrates significant incident resolution time improvements with exceptional correlation accuracy [7, 8]
Prompt-Based Interaction	Natural language processing engines trained on DNS-specific terminology and debugging patterns	Eliminates traditional command syntax requirements; processes extensive domain-specific failure events rapidly with ranked failure analysis, statistical confidence intervals, and temporal correlation patterns
Conversational AI Layer	Multi-dimensional indexing algorithms processing tenant-specific failure patterns with advanced anonymization	Reduces cognitive burden during high-stress incidents; substantially reduces debugging command formulation errors; supports numerous distinct natural language patterns for DNS-specific troubleshooting scenarios while maintaining tenant data privacy

Table 3: AI-Enhanced DNS Debugging Through MCP Integration: Deployment Options and Capabilities [7, 8]

5. Evaluation: Case Scenarios

The effectiveness of DRIS has been demonstrated through several real-world scenarios across enterprise-scale DNS infrastructures spanning multiple geographical regions and processing substantial volumes of DNS queries daily. Performance validation involved comprehensive testing across diverse operational environments, including multi-tenant cloud platforms serving extensive active tenant populations and distributed resolver networks handling substantial peak traffic loads during critical business periods.

Incident triage scenarios have proven particularly valuable in demonstrating DRIS capabilities during complex regional timeout events affecting numerous data centers across continental regions. Such incidents typically impact substantial concurrent user session populations, with DNS resolution failures cascading through numerous distinct application services. DRIS processes extensive volumes of distributed log data from numerous DNS service endpoints, rapidly identifying root causes such as configuration drift in forwarding rules that affect significant percentages of recursive resolver queries in impacted regions [9]. Traditional debugging approaches require manual correlation of logs from multiple service tiers, with estimated resolution times extending significantly beyond acceptable incident response thresholds based on historical incident patterns. The dramatic improvement in response time demonstrates the system's practical value in production environments, with incident cost avoidance representing substantial operational savings based on standard service level agreement penalties and operational impact assessments.

Regression analysis capabilities enable a comprehensive comparison of recursive resolution success rates before and after the deployment of critical resolver updates affecting extensive production resolver clusters. Pre-deployment baseline analysis processes substantial volumes of DNS queries over extended observation periods, establishing comprehensive resolution success rate baselines with detailed response time measurements. Post-deployment monitoring identifies subtle degradations in resolution success rates, with timeout anomalies increasing substantially in specific geographic regions serving significant active user populations. The regression analysis correlates deployment timestamps with performance degradation patterns across extensive, distinct query types, identifying specific query name patterns that trigger performance regressions.

DRIS processes comprehensive correlation matrices spanning substantial individual DNS events, isolating regressions to specific record processing logic that introduces latency spikes for affected queries. This capability proves invaluable for change management and deployment validation processes, with regression detection occurring rapidly after deployment completion compared to traditional monitoring approaches that require extended periods for comprehensive regression identification [10]. The rapid regression detection enables prompt rollback procedures that minimize service impact compared to extended degraded performance periods required for manual detection and remediation.

These case studies highlight DRIS's ability to transform incident response from a reactive, time-intensive process into a structured, efficient analysis workflow. Quantitative analysis of debugging workflows demonstrates substantial time-to-resolution improvements across documented incidents spanning extended periods of production deployment. The system's correlation capabilities process multi-dimensional failure patterns across distributed service architectures, analyzing extensive potential interaction points while maintaining exceptional accuracy in root cause identification.

Additional performance metrics indicate consistent analytical capabilities across diverse failure scenarios, including timeout cascades affecting numerous service tiers, domain-specific failures impacting substantial unique domains, and tenant isolation breaches affecting multi-tenant DNS infrastructures. DRIS maintains optimal query response times while processing analytical workloads that span substantial volumes of historical telemetry data, enabling rapid correlation analysis during high-pressure incident escalation scenarios.

Evaluation Scenario	System Capabilities and Process	Operational Benefits and Outcomes
Incident Triage Scenarios	Processes extensive volumes of distributed log data from numerous DNS service endpoints during complex regional timeout events; rapidly identifies root causes such as configuration drift in forwarding rules affecting significant percentages of recursive resolver queries [9]	Demonstrates dramatic improvement in response time with substantial operational cost avoidance; transforms reactive debugging from manual correlation across multiple service tiers to structured, efficient analysis workflows
Regression Analysis Capabilities	Enables comprehensive comparison of recursive resolution success rates before and after critical resolver updates; correlates deployment timestamps with performance degradation patterns across extensive, distinct query types, isolating regressions to specific record processing logic [10]	Proves invaluable for change management and deployment validation processes; enables rapid rollback procedures that minimize service impact compared to extended degraded performance periods required for manual detection
Multi-Dimensional Performance Analysis	Maintains consistent analytical capabilities across diverse failure scenarios, including timeout cascades, domain-specific failures, and tenant isolation breaches; processes correlation matrices spanning substantial individual DNS events with exceptional accuracy in root cause identification	Achieves substantial time-to-resolution improvements across documented incidents; maintains optimal query response times while processing analytical workloads spanning substantial volumes of historical telemetry data during high-pressure incident escalation scenarios

Table 4: Operational Effectiveness Demonstration: DRIS Case Studies and Performance Metrics [9, 10]

6. Additional Capabilities

DRIS addresses critical debugging gaps, yet faces limitations that present opportunities for future enhancement across multiple operational dimensions. Current deployment analysis indicates that substantial percentages of enterprise DNS operations require additional integration capabilities to achieve comprehensive incident response coverage, with performance benchmarks suggesting significant potential improvements in mean time to mitigation through enhanced system capabilities.

Integration with self-service dashboards for Directly Responsible Individuals represents a significant enhancement opportunity, with current enterprise environments supporting extensive DRI populations across distributed operational teams. Self-service dashboard integration would enable real-time access to DRIS analytical capabilities for numerous concurrent users, substantially reducing dependency on specialized debugging expertise [11]. Performance projections indicate that dashboard integration could process substantial DRI query volumes while maintaining optimal response times for standard DNS analysis requests. The integration architecture would support comprehensive role-based access controls across numerous distinct permission levels, enabling secure self-service debugging for tenant-specific DNS issues affecting extensive individual tenant identifier populations. Dashboard analytics suggest substantial operational cost reductions through reduced escalation overhead and improved first-level resolution rates for routine DNS infrastructure issues.

Expanded telemetry feeds to enrich context and provide more comprehensive analysis capabilities would substantially increase DRIS data ingestion capacity from current processing volumes to significantly enhanced daily processing across extended telemetry sources. Enhanced telemetry integration would incorporate data from numerous additional monitoring endpoints, including application performance metrics, network topology changes, and security event correlations. The expanded telemetry architecture would process comprehensive correlation matrices involving substantially more potential interaction points compared to current interaction point limitations. Performance modeling indicates significant accuracy improvements in root cause identification when incorporating enriched telemetry context, with correlation analysis timeframes reducing substantially for complex multi-service debugging scenarios. Extended telemetry integration would support historical analysis spanning extended retention periods across substantial unique domain tracking entries.

Multi-agent orchestration capabilities, where agents critique, correlate, and propose remediations autonomously, represent sophisticated enhancement opportunities with substantial potential for automation of routine DNS debugging workflows. Multi-agent frameworks would coordinate between DRIS analytical agents and complementary infrastructure agents, processing autonomous debugging scenarios across extensive, distinct failure pattern categories. Orchestration engines would maintain concurrent agent populations of numerous specialized debugging agents, each optimized for specific DNS service layer analysis. Performance projections indicate autonomous root cause identification capabilities reaching exceptional accuracy across complex distributed system failures, with agent collaboration substantially reducing human intervention requirements.

Proactive AI-driven monitoring using DRIS queries to detect anomalies preemptively would transform incident response paradigms through predictive analytics capabilities processing substantial volumes of real-time DNS telemetry. Proactive monitoring algorithms would analyze historical patterns across extended baseline periods, detecting deviation thresholds exceeding normal operational parameters. The AI-driven monitoring system would generate early warning alerts with exceptional accuracy in predicting DNS infrastructure failures, substantial time periods before critical threshold breaches [12]. Predictive analytics would process trend analysis across extensive, distinct DNS service metrics, maintaining minimal false positive rates while achieving comprehensive coverage of potential failure scenarios.

When paired with monitoring agents, DRIS enables proactive monitoring capabilities, shifting from reactive incident response to predictive maintenance and early warning systems. Integration testing demonstrates that proactive monitoring capabilities substantially reduce unplanned DNS outages, with predictive maintenance workflows preventing significant percentages of critical service interruptions. The combined proactive monitoring and DRIS analytical platform maintains exceptional service availability while processing predictive analysis workloads spanning extensive geographical regions and tenant environments.

Conclusion

DNS failures continue to represent some of the most challenging operational incidents for enterprise infrastructure teams. The DNS Resolution Insight System establishes a transformative paradigm for DNS infrastructure debugging by unifying disparate log sources into coherent analytical frameworks while integrating advanced AI-driven interaction capabilities. The system's Model Context Protocol integration enables engineers to query complex DNS telemetry through natural language interfaces, fundamentally altering incident response workflows from reactive log hunting into structured, collaborative debugging processes. The DRIS architecture provides both incident response and proactive anomaly detection as a result of standardized analytic interfaces that work with pre-existing enterprise monitoring ecosystems. Furthermore, the scalability of the platform in various deployment configurations-from local development environments to geographically distributed multi-tenant services-will allow for the capability to work regardless of organizational structure. Increased functionality that includes self-service dashboard integrations, more telemetry feeds, multi-agent orchestration, and proactive monitoring with AI provides ample opportunity for improving DNS operational excellence. Future development will focus on deeper integration with multi-agent observability frameworks to create comprehensive, intelligent DNS monitoring and debugging ecosystems. The synthesis of traditional network troubleshooting methodologies with contemporary AI-driven correlation techniques positions DRIS as a foundational capability for maintaining DNS infrastructure reliability in increasingly complex distributed computing environments.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Dennis Tatang, et al., "Large-Scale Analysis of Infrastructure-Leaking DNS Servers," ResearchGate, 2019. [Online]. Available: https://www.researchgate.net/publication/333675169_Large-Scale_Analysis_of_Infrastructure-Leaking_DNS_Servers
[2] Rahul Kumar Dubey and Prof. (Dr.) Rajeev Yadav, "Automated Network Configuration and Troubleshooting Using Natural Language Processing," International Journal of Advanced Research in Science and Engineering, 2024. [Online]. Available: https://www.ijarse.com/images/fullpdf/1724827252_B2406.pdf
[3] Sarika Sharma, "The Role of Behavioral Machine Learning in Detecting Network Anomalies at Scale," Fidelis Security, 2025. [Online]. Available: https://fidelissecurity.com/threatgeek/network-security/network-behavior-anomaly-detection-at-scale/
[4] Mohammad (Matt) Namvarpour and Afsaneh Razi, "The Art of Talking Machines: A Comprehensive Literature Review of Conversational User Interfaces," ACM Digital Library, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3719160.3736621

[5] Design Gurus, "How would you design a log aggregation system that collects and indexes logs from millions of servers?" [Online]. Available: https://www.designgurus.io/answers/detail/how-would-you-design-a-log-aggregation-system-that-collects-and-indexes-logs-from-millions-of-servers

[6] Aniss Maghsoudlou, et al., "FlowDNS: Correlating Netflow and DNS Streams at Scale," CoNEXT '22, 2022. [Online]. Available: https://olivergasser.net/papers/maghsoudlou2022flowdns.pdf

[7] Neo Miguel, "Model context protocol integration patterns," BytePlus, 2025. [Online]. Available: https://www.byteplus.com/en/topic/541370?title=model-context-protocol-integration-patterns

[8] Mubeen Wasif and David Tunkel, "Multi-Agent Collaboration in AI: Enhancing Software Development with Autonomous LLMs," ResearchGate, 2025. [Online]. Available: https://www.researchgate.net/publication/388834996_Multi-Agent_Collaboration_in_AI_Enhancing_Software_Development_with_Autonomous_LLMs

[9] Ankur Mahida, "Real-Time Incident Response and Remediation-A Review Paper," ResearchGate, 2023. [Online]. Available: https://www.researchgate.net/publication/379793503_Real-Time_Incident_Response_and_Remediation-A_Review_Paper

[10] Arun Pamulapati, "Automating ML, Scoring, and Alerting for Detecting Criminals and Nation States Through DNS Analytics," Databricks, 2022. [Online]. Available: https://www.databricks.com/blog/2022/08/02/automating-ml-scoring-and-alerting-for-detecting-criminals-and-nation-states-through-dns-analytics.html

[11] Keyur Patel, "Self-Service Analytics: Strategies & Best Practices," IT Path Solutions, 2025. [Online]. Available: https://www.itpathsolutions.com/self-service-analytics-strategies-and-best-practices/

[12] Oluwatosin Aramide, "Predictive Network Maintenance and Anomaly Detection with AI," ResearchGate, 2025. [Online]. Available: https://www.researchgate.net/publication/393744129_Predictive_Network_Maintenance_and_Anomaly_Detection_with_AI