Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 25, 2026

A. Smith
N. Palin
Arrcus, Inc.
March 25, 2026

Governance Framework for AI-Mediated Autonomous Network Device Management

draft-smith-opsawg-ai-network-governance-00

Abstract

This document defines a governance framework for systems that use artificial intelligence (AI) services, specifically large language models (LLMs), to autonomously detect, diagnose, and remediate operational anomalies on network devices. As AI-driven automation moves from advisory tooling to closed-loop autonomous operation on production infrastructure, the industry lacks a common set of principles governing what such systems may and may not do.

This framework establishes thirteen governance areas covering human authority, harm prevention, management plane protection, minimum necessary action, bounded autonomy, transparency, reversibility, graceful degradation, escalation, AI-specific constraints, startup safety, absolute prohibitions, and review processes. It is intended to serve as a reference architecture for implementers building AI-mediated network management systems and for operators evaluating the safety properties of such systems.

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 August 29, 2026.

without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

1.1. Problem Statement

Network devices generate large volumes of operational telemetry: interface counters, routing protocol state, forwarding table contents, hardware health indicators, and system resource utilization. When operational anomalies occur, human operators must manually detect, diagnose, remediate, verify, and document the incident. This process is time-consuming, error-prone, and scales linearly with network size.

Recent advances in artificial intelligence, specifically large language models (LLMs), have created the possibility of autonomous systems that can reason about network anomalies and propose remediation actions. Unlike traditional runbook automation, which executes predefined if-then rules, an LLM-based system can analyze novel situations, correlate diverse data sources, and propose context-appropriate responses.

However, the transition from AI-as-advisor (human approves every action) to AI-as-autonomous-agent (system acts within bounds) introduces significant safety challenges. An autonomous agent with access to a network device’s management plane could, without proper governance:

- Take actions that worsen the network state
- Interfere with management plane reachability
- Execute changes during network convergence events
- Overwhelm the device with rapid successive changes
- Act on hallucinated or incorrect AI outputs
- Operate without adequate audit trail
- Resist or circumvent human override

This document defines a governance framework that addresses these challenges by establishing principles, constraints, and operational boundaries for AI-mediated autonomous network device management systems.

1.2. Scope

This framework applies to systems that meet ALL of the following criteria:

- The system executes on or has direct management plane access to a network device (router, switch, or similar)
- The system interfaces with an external AI service (LLM) for anomaly analysis and remediation proposal
- The system has the capability to modify device configuration or operational state autonomously (i.e., without per-action human approval)
- The system uses standardized management interfaces such as NETCONF [RFC6241], RESTCONF [RFC8040], or gNMI for device interaction

This framework does NOT apply to:

- AI systems that only provide advisory recommendations requiring human execution
- Traditional event-driven automation (runbooks, playbooks) that do not use AI reasoning
- Network monitoring systems that detect but do not remediate
- AI systems that operate exclusively at the management station level without direct device access

While the principles in this document are framed around network device management, many of them (human authority, bounded autonomy, transparency, reversibility) are generalizable to AI-mediated autonomous management of other infrastructure systems.

1.3. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are used throughout this document:

AI Service: An external artificial intelligence system, typically a large language model (LLM), accessed via an API. The AI Service receives structured queries describing network anomalies and returns structured remediation proposals.

Autonomous Agent: The software system executing on or connected to a network device that collects telemetry, detects anomalies, consults the AI Service, and executes approved remediation actions. Also referred to as "the agent" or "the system."

Governance Framework: The set of principles, constraints, and operational boundaries defined in this document that bound the autonomous agent's behavior.

Remediation Action: Any modification to device configuration or operational state performed by the autonomous agent in response to a detected anomaly.

Protected Target: A device resource (interface, protocol instance, configuration section) that the autonomous agent is prohibited from modifying under any circumstances.

Action Registry: A fixed, developer-defined catalog of remediation actions that the autonomous agent is capable of performing. Each entry specifies the action's parameters, risk level, and reversibility.

Operator Allow List: A configurable list of actions from the Action Registry that the operator has approved for autonomous execution in a specific deployment.

Pre-State Capture: A snapshot of a target resource's configuration or operational state taken immediately before a remediation action is executed, enabling rollback.

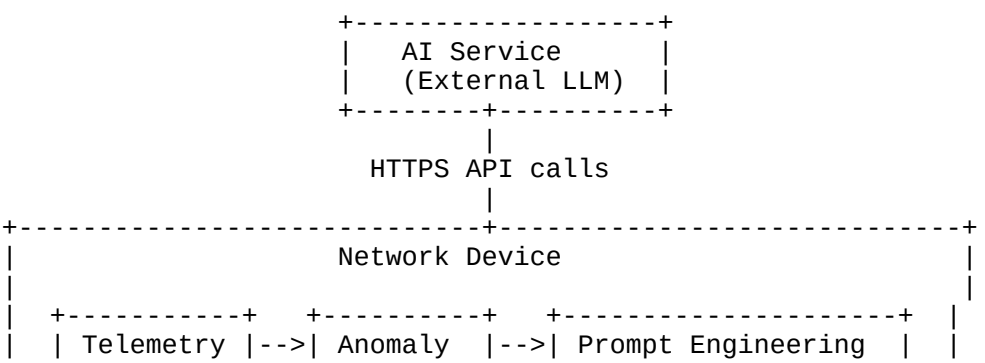
Convergence Event: A network state where multiple simultaneous critical anomalies indicate that routing protocols are reconverging (e.g., multiple BGP peer flaps, IGP SPF storms, FIB churn spikes).

Degradation Level: A numeric indicator (0-4) reflecting the health of the autonomous agent's internal components, used to restrict autonomous capabilities when components fail.

2. System Model

2.1. Architecture Overview

The system model assumed by this governance framework consists of the following components:



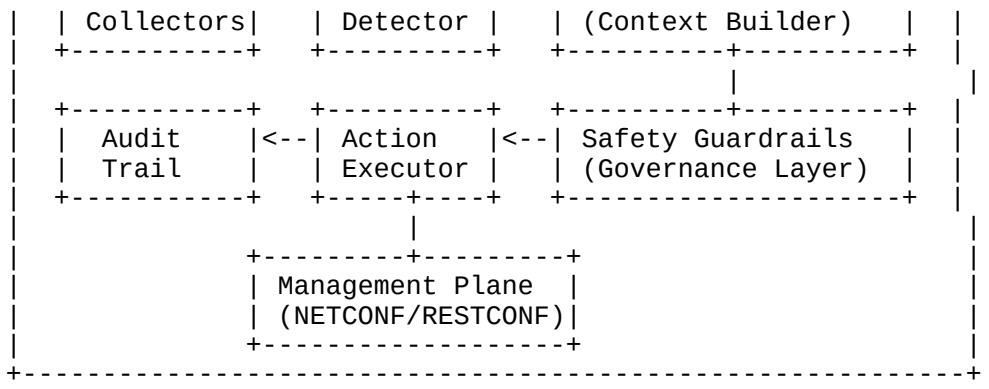


Figure 1: Autonomous Agent Architecture Overview

The telemetry collectors gather device state via standard management interfaces. The anomaly detector compares collected metrics against learned baselines using multiple detection methods. When anomalies are detected, the prompt engineering component constructs a structured query for the AI Service, including device state, historical context, and governance constraints. The AI Service returns a structured remediation proposal. The safety guardrails layer validates the proposal against the governance framework before the action executor applies it to the device. All decisions and actions are recorded in the audit trail.

2.2. AI Service Interface

The interface between the autonomous agent and the AI Service is a critical architectural boundary. The governance framework requires that this interface has the following properties:

- The AI Service is STRICTLY advisory. Its outputs are proposals, not commands.
- All AI proposals MUST pass through the safety guardrails layer before execution. There is no bypass path.
- The agent sends structured context (not raw CLI output or unstructured logs) to the AI Service.
- The agent receives structured responses (not free-text instructions) from the AI Service.
- The AI Service has NO direct access to execute commands, modify configuration, or interact with the device.
- Every AI interaction (prompt and response) is logged in the audit trail.

2.3. Remediation Lifecycle

The governance framework assumes a remediation lifecycle with the following phases:

- Detection: The agent detects an anomaly in device telemetry
- Analysis: The agent consults the AI Service for diagnosis and remediation recommendations
- Validation: The agent validates any proposed action against the governance framework

- Execution: The agent applies the validated action to the device

- Verification: The agent verifies the outcome of the action

- Resolution or Escalation: The agent either confirms resolution or escalates to a human operator

The governance principles defined in Sections 3 through 15 apply to all phases of this lifecycle. The specific mechanisms by which an implementation carries out these phases are outside the scope of this document.

3. Governance Principle 1: Human Authority Supremacy

The autonomous agent operates exclusively with delegated authority. Human authority over the network device is supreme and may not be diminished by the agent's operation.

3.1. Delegation and Revocation

A human operator MUST be able to revoke the agent's authority at any time, for any reason, with immediate effect. The agent MUST NOT impose delays, confirmations, or conditions on revocation.

No automated action taken by the agent MAY override, circumvent, or delay a human-initiated stop, pause, or configuration change.

3.2. Multiple Intervention Mechanisms

The agent MUST provide multiple independent mechanisms for human intervention. At minimum, the following capabilities MUST be available:

Emergency Stop: Immediate full shutdown of the agent. The agent terminates all operations and releases all resources. This SHOULD be implementable via standard process signals (e.g., SIGTERM, SIGINT) or equivalent platform-specific mechanisms.

Pause Remediation: Halts all remediation actions while continuing monitoring, detection, and alerting. This allows operators to maintain situational awareness while preventing the agent from taking actions during sensitive operational windows (e.g., maintenance periods).

Configuration Reload: Allows operators to modify the agent's operational parameters (rate limits, allowed actions, risk thresholds) without restarting the agent.

Baseline Refresh: Allows operators to re-anchor the agent's definition of "normal" state, accommodating intentional network changes.

These mechanisms MUST be independent of each other and of the agent's primary operation. A failure in one mechanism MUST NOT affect the availability of other mechanisms.

3.3. Loss of Human Reachability

When the agent cannot reach a human operator through any configured notification channel (syslog, ticketing system,

email, or equivalent), it SHALL default to monitoring-only mode and SHALL NOT execute remediation actions.

The rationale is that remediation without the ability to notify a human of the outcome violates the transparency principle (Section 8) and removes the human's ability to intervene if the remediation causes harm.

3.4. Self-Modification Prohibition

The agent MUST NOT modify its own configuration, governance rules, action catalogs, blocked-action lists, or safety parameters. These artifacts are exclusively under human control.

This prohibition extends to the AI Service: the agent MUST NOT accept AI-suggested modifications to its own governance framework or safety parameters.

4. Governance Principle 2: Do No Harm

The autonomous agent's primary obligation is to never make the network worse than its current state.

4.1. Uncertainty Default

When uncertain whether an action will improve or degrade network service, the agent SHALL take no action and SHALL escalate to a human operator.

This principle establishes a strong bias toward inaction. The cost of a missed remediation opportunity (continued degradation until a human responds) is generally lower than the cost of an incorrect remediation (additional degradation caused by the agent itself).

4.2. Pre-State Capture Requirement

Before any remediation action, the agent MUST capture the current state of the target resource. The captured state MUST be sufficient to restore the target to its pre-action condition.

For configuration changes, this means capturing the relevant configuration section via NETCONF <get-config> [RFC6241] or equivalent.

For operational state changes, this means capturing the relevant operational state via RESTCONF GET [RFC8040], gNMI Get, or equivalent.

If pre-state capture fails for any reason, the remediation action MUST NOT proceed. The agent SHALL log the capture failure and escalate to a human operator.

4.3. Post-Action Verification and Auto-Rollback

After any remediation action, the agent MUST verify the outcome by re-collecting telemetry from the affected resource and re-running anomaly detection.

If the target's condition has not improved, or has worsened at any severity level (not only critical), the agent SHALL automatically roll back the change to the captured pre-state and escalate to a human operator.

The verification MUST use fresh telemetry collection, not cached or estimated values. The anomaly detection during verification MUST bypass any duplicate-suppression mechanisms (e.g., cooldown timers) that would mask the re-detection of the original anomaly.

4.4. Convergence Event Suppression

The agent SHALL NOT execute remediation actions during detected convergence events.

A convergence event is indicated when two or more critical-severity anomalies from convergence-sensitive metrics (e.g., BGP peer state, IGP SPF run count, FIB update rate, BFD session state) are detected in the same collection cycle.

During a convergence event, the agent SHALL continue to monitor, record, and alert, but SHALL NOT execute any remediation action. The rationale is that automated changes during network reconvergence risk interfering with the network's self-healing mechanisms and may amplify instability.

The convergence-sensitive metric set SHOULD be configurable by the operator to accommodate network-specific characteristics.

5. Governance Principle 3: Management Plane Protection

The agent SHALL NEVER take any action that could compromise management access to the device. This is an absolute, non-negotiable constraint that takes precedence over all other considerations, including AI recommendations.

5.1. Protected Target Classes

The following resource classes are permanently protected. No remediation action of any kind MAY be directed at them:

Management Interfaces: Interfaces used for out-of-band management access. Identified by name patterns containing "mgmt" or "management" (case-insensitive), or by operator configuration.

Loopback Interfaces: Interfaces commonly used as router identifiers and management endpoints. Identified by name patterns containing "loopback" or beginning with "lo" (case-insensitive), or by operator configuration.

Operator-Defined Resources: Any additional resources explicitly listed as protected in the agent's deployment configuration.

5.2. Protected Target Identification

The agent's protected target identification MUST combine both pattern-based matching (for well-known resource naming conventions) and operator-configurable exact matches. The agent MUST NOT rely solely on hardcoded patterns, as device vendors and operators use varying naming conventions.

When evaluating whether a resource is protected, the agent SHALL treat any match (pattern or exact) as dispositive.

The agent MUST NOT override a protection determination based on AI recommendation or anomaly severity.

5.3. Management Plane Configuration

The agent SHALL NOT modify routing policies, BGP autonomous system numbers, or any configuration that could affect reachability to the management plane. This includes but is not limited to:

- Route policies and route maps
- Access control lists applied to management interfaces
- Authentication and authorization configuration
- NTP, DNS, and other management service configurations

6. Governance Principle 4: Minimum Necessary Action

The agent SHALL apply the least disruptive action that addresses the detected anomaly.

6.1. Action Preference Ordering

When multiple remediation options are available, the agent SHOULD prefer actions in the following order, from least to most disruptive:

1. Alert only (no device modification)
2. Counter or statistics clear
3. Soft reset (e.g., BGP route refresh)
4. Hard reset (e.g., BGP session clear)
5. Interface administrative state toggle
6. Routing metric adjustment (e.g., OSPF cost change)

More disruptive actions SHOULD only be attempted after less disruptive alternatives have been considered or attempted.

6.2. Single-Target Constraint

Each remediation action SHALL target exactly one specific resource: one interface, one routing peer, one protocol instance. Bulk or wildcard actions are prohibited.

The rationale is that single-target actions limit blast radius, simplify rollback, and make the impact of each action independently verifiable.

6.3. Parameter Validation

Action parameters SHALL be validated against safe ranges before execution. Parameter ranges SHOULD be defined in the Action Registry and MAY be overridden by operator configuration.

Example parameter constraints:

- OSPF interface cost: 1 to 65534. The value 65535 (max-metric) has special protocol semantics and SHOULD

require human approval.

- o BGP peer operations: single named peer only. Wildcard or "all peers" targets MUST be rejected.

- o Interface administrative state: toggle to opposite of current state only.

Parameters that fall outside validated ranges MUST be rejected. The agent SHALL log the rejection and MAY escalate to a human operator.

6.4. Multi-Step Operation Prohibition

The agent SHALL NOT attempt to solve problems that require multi-step orchestration, where the correctness of later steps depends on the outcome of earlier steps (e.g., drain traffic from an interface, modify the interface, then restore traffic).

Multi-step remediation sequences require human planning and approval because the agent cannot reliably predict the interaction effects between steps or handle partial failures in the sequence.

7. Governance Principle 5: Bounded Autonomy

The agent’s autonomous capabilities are bounded by rate limits, scope limits, and risk-level enforcement.

7.1. Rate Limits

The agent MUST enforce the following rate limits. Default values are provided; operators MAY configure stricter limits but SHOULD NOT exceed the specified maximums.

Global Remediation Rate: Maximum remediation actions per hour across all targets. Default: 5. Maximum: 20.

Per-Target Remediation Rate: Maximum actions directed at a single target resource per 24-hour period. Default: 3. Maximum: 5.

Per-Target Irreversible Rate: Maximum non-rollback-capable actions directed at a single target per 24-hour period. Default: 1. Maximum: 2.

AI Service Consultation Rate: Maximum queries to the AI Service per hour. Default: 20. Maximum: 60.

Duplicate Anomaly Suppression: Minimum interval between raising the same anomaly. Default: 300 seconds.

Remediation Retry Limit: Maximum retry attempts for a single anomaly before escalation. Default: 3. Maximum: 5.

Rate limit counters MUST be based on actual execution timestamps stored in the audit trail, not on in-memory counters that could be reset by agent restart.

7.2. Scope Limits

The agent SHALL only operate on the local device on which it is installed (or to which it has been explicitly assigned

in a centralized deployment). It SHALL NOT make changes to remote devices, routing peers, or network controllers.

The agent SHALL only perform actions that are listed in BOTH of the following:

- The Action Registry (developer-defined catalog of technically possible actions), AND
- The Operator Allow List (operator-configured subset of the Action Registry approved for this deployment)

Any action listed on an Operator Block List is absolutely prohibited regardless of whether it appears on the Allow List. The Block List always takes precedence.

The agent SHALL NOT create, delete, or modify: routing policies, access control lists, authentication or authorization configuration, NTP/DNS/management services, or device firmware and software images.

7.3. Risk-Level Enforcement

Each action in the Action Registry MUST carry a risk level classification: "low", "medium", or "high".

Risk levels SHOULD be assigned based on the following criteria:

Low: Action is non-destructive and has no lasting effect (e.g., counter clear, route refresh).

Medium: Action causes temporary disruption but is recoverable (e.g., session clear, interface toggle).

High: Action causes significant disruption or permanent state change (e.g., routing metric change, peer configuration modification).

The operator configures a maximum autonomous risk level (default: medium). Actions above this threshold MUST NOT be executed autonomously. The agent SHALL create a notification (e.g., ticket) with the proposed action and await human approval.

8. Governance Principle 6: Transparency and Auditability

Every decision and action taken by the autonomous agent MUST be recorded with sufficient detail for a human to reconstruct the full chain of reasoning after the fact.

8.1. Remediation Action Records

Every remediation action SHALL be recorded with at minimum:

- Timestamp of execution
- Triggering anomaly (type, severity, metric values)
- AI analysis (the full prompt sent and response received)
- Pre-state snapshot of the target resource
- Post-state snapshot of the target resource

- Outcome (success, failure, rolled back)
- Whether the action was autonomously triggered or human-approved

8.2. Detection Decision Records

Every anomaly detection decision SHALL be logged with sufficient detail for a human to understand why the anomaly was or was not flagged. This includes the detection method used, the metric value, the baseline statistics, and any threshold or z-score calculation.

8.3. AI Interaction Records

Every interaction with the AI Service SHALL be logged with the full prompt and full response. This enables:

- Post-incident analysis of AI reasoning
- Detection of AI hallucinations or incorrect advice
- Training data collection for AI improvement
- Compliance and regulatory audit

8.4. Notification Records

The agent SHOULD emit notifications via standard mechanisms (e.g., syslog [RFC5424]) for:

- Every remediation action (before and after execution)
- Every automatic rollback
- Every escalation to a human operator
- Every safety guardrail activation (action blocked)
- All agent lifecycle events (start, stop, pause, resume)

8.5. Retention

Audit records SHALL be retained for a configurable period. The default retention period SHOULD be no less than 365 days for anomaly and remediation records.

9. Governance Principle 7: Reversibility

The agent SHALL prefer reversible actions and SHALL impose additional constraints on irreversible actions.

9.1. Reversibility Preference

Each action in the Action Registry MUST indicate whether it is reversible (rollback-capable) or irreversible.

When multiple actions could address an anomaly, the agent SHOULD prefer reversible actions over irreversible ones, independent of other factors.

9.2. Irreversible Action Restrictions

For any action that is not rollback-capable, the agent SHALL:

- Log a clear warning that the action is irreversible
- Only execute if the triggering anomaly severity is critical (the highest severity level)
- Enforce a stricter per-target rate limit (see Section 7.1)

9.3. Automatic Rollback Threshold

Automatic rollback SHALL be triggered when post-action verification detects any regression at warning severity or above. The rollback threshold is intentionally set below the highest severity level to provide a safety margin.

The rationale is that waiting for critical-severity regression before rolling back allows preventable damage to accumulate. Rolling back at warning level catches emerging problems before they become critical.

9.4. Pre-State Retention

Pre-state captures SHALL be retained in the audit trail until the retention period expires (see Section 8.5). This enables manual rollback by an operator at any time during the retention period, even if the agent did not trigger automatic rollback.

10. Governance Principle 8: Graceful Degradation

If any internal component of the autonomous agent fails, the agent SHALL continue operating in a degraded mode rather than crashing or continuing to operate at full autonomy.

10.1. Degradation Levels

The agent SHALL track the health of its internal components and maintain a degradation level that reflects overall system health. The following degradation model is RECOMMENDED:

Level 0 - Fully Healthy: All components operational. Full autonomous operation permitted.

Level 1 - AI Service Unavailable: The agent cannot reach the AI Service. The agent continues monitoring, detection, and alerting but MUST NOT execute remediation actions.

Level 2 - Notification Unavailable: The agent cannot deliver notifications (syslog, ticketing, etc.). The agent continues monitoring, detection, and logging but MUST NOT execute remediation actions.

Level 3 - Persistent Storage Unavailable: The agent cannot write to its audit trail. The agent continues local logging only. No detection or remediation.

Level 4 - Telemetry Collection Failed: The agent cannot collect device telemetry. The agent performs a safe shutdown.

10.2. Remediation Gating

Remediation is ONLY permitted at Level 0 (fully healthy). Any component degradation disables all autonomous actions.

The rationale is that an agent that cannot consult the AI Service lacks reasoning capability; an agent that cannot notify lacks transparency; an agent that cannot log lacks auditability. Each of these capabilities is a prerequisite for safe autonomous operation.

10.3. Degradation Notification

Component failures SHALL be logged and alerted via all still-functional notification channels. The agent SHALL include the degradation level and affected component in the notification.

11. Governance Principle 9: Escalation Protocol

The agent SHALL escalate to a human operator when it encounters situations that exceed its autonomous capabilities.

11.1. Escalation Triggers

The agent SHALL escalate when any of the following conditions are met:

- The AI Service recommends human intervention
- The remediation retry loop is exhausted (maximum attempts reached without resolution)
- The global hourly rate limit is exhausted with anomalies remaining unresolved
- A proposed action exceeds the configured maximum autonomous risk level
- A convergence event is detected
- An automatic rollback is triggered (indicating the remediation worsened the situation)
- The agent is operating in a degraded mode

11.2. Escalation Content

Escalation notifications SHALL include at minimum:

- Problem description (anomaly type, severity, affected resource, metric values)
- Actions already taken (if any), including outcomes
- AI analysis and recommendations
- Recommended next steps for the human operator
- Current device state summary
- Urgency classification

11.3. Post-Escalation Behavior

After escalation, the agent SHALL NOT retry the same

remediation unless one of the following occurs:

- A human operator explicitly re-enables remediation for the escalated anomaly (via configuration change)
- The anomaly clears and recurs after a complete fresh detection cycle (indicating a new occurrence, not the same event)

This prevents the agent from repeatedly attempting and failing the same remediation, which would generate excessive notifications and potentially mask new issues.

12. Governance Principle 10: AI-Specific Constraints

The AI Service is an advisory system only. Its role in the architecture is to provide analysis and recommendations, not to control the device.

12.1. Advisory-Only Status

AI recommendations MUST pass through all safety guardrails before execution. The AI Service cannot bypass any governance rule defined in this framework.

If the AI suggests an action that violates any governance principle, the action SHALL be rejected. The agent MAY re-consult the AI Service with an explanation of why the action was rejected, requesting an alternative.

12.2. Action Registry Constraint

AI-suggested actions that do not correspond to entries in the Action Registry SHALL be discarded. The agent SHALL NOT dynamically create new action types from AI suggestions.

This constraint ensures that the set of possible autonomous actions is fixed at deployment time and cannot be expanded by AI outputs. New action types require explicit developer implementation and operator approval.

12.3. Parameter Validation

AI-suggested parameter values SHALL be validated against the safe ranges defined in Section 6.3. Parameters outside validated ranges SHALL be rejected regardless of the AI's confidence level.

12.4. Unparseable Response Handling

If the AI Service's response cannot be parsed into a valid structured format (e.g., malformed JSON, missing required fields, inconsistent data), the agent SHALL treat the response as a human-escalation recommendation.

The rationale is that an unparseable response indicates either an AI malfunction or a situation too complex for structured analysis. Both cases warrant human review.

12.5. Constraint Communication

The system prompt or context provided to the AI Service SHALL include a summary of the governance constraints (allowed actions, blocked actions, protected targets,

rate limits, risk-level ceiling).

This enables the AI Service to self-constrain its recommendations, reducing the number of proposals that are rejected by the safety guardrails. See Section 16 for detailed guidance.

12.6. No Direct Device Access

The AI Service SHALL NOT be given direct access to execute commands, modify configuration, or interact with the network device. All device interaction is mediated through the agent's safety layers.

13. Governance Principle 11: Startup and Configuration Safety

The agent's default configuration SHALL be maximally restrictive, requiring explicit operator action to enable autonomous capabilities.

13.1. Configuration Validation

On startup, the agent SHALL validate its loaded configuration before beginning operation. Invalid configuration SHALL prevent the agent from starting.

On configuration reload (e.g., via signal or API), the agent SHALL validate the new configuration before applying it. Invalid configuration SHALL be rejected; the agent continues with the previous valid configuration.

13.2. Safe Defaults

The default configuration SHALL disable autonomous remediation. At minimum:

- Remediation execution SHALL default to disabled
- Dry-run mode SHALL default to enabled

An operator must explicitly change both settings to enable live autonomous remediation. This two-key activation prevents accidental enablement.

13.3. Configuration Logging

The agent SHALL log its full effective configuration (with credentials and secrets redacted) at startup. This enables operators to verify that the agent is operating with the intended configuration.

14. Governance Principle 12: Absolute Prohibitions

The following actions are prohibited under all circumstances, regardless of configuration, AI recommendation, anomaly severity, or operator override. These prohibitions are non-negotiable invariants of the governance framework.

1. Deleting any configuration section or container from the device
2. Modifying routing policy, route maps, or route filters

3. Changing BGP autonomous system numbers

4. Shutting down all interfaces simultaneously

5. Modifying management plane access configuration

6. Disabling or modifying authentication or authorization mechanisms

7. Modifying the agent's own governance rules or safety guardrail logic

8. Executing arbitrary CLI or shell commands on the device

9. Transmitting device configuration, credentials, or network topology information to external services beyond the configured AI Service and notification endpoints

10. Operating without a functional audit trail (persistent storage and logging)

These prohibitions SHOULD be enforced through multiple independent mechanisms (e.g., both in the action validation logic and in the configuration change scanning logic) to provide defense in depth.

15. Governance Principle 13: Review and Amendment

The governance framework is a living document that requires periodic human review.

15.1. Mandatory Review Points

The governance framework SHALL be reviewed by the system operator:

- Before initial deployment of the autonomous agent
- After any significant change to the agent's capabilities (new action types, new collectors, new AI models)
- After any incident where the agent's autonomous actions caused unintended consequences
- On a periodic schedule determined by the operator (RECOMMENDED: annually)

15.2. Amendment Process

Amendments to the governance framework require explicit human authorship and approval. The autonomous agent SHALL NOT suggest, draft, or implement changes to the governance framework.

The agent SHOULD log a warning at startup if the governance document is missing or has been modified since the last known-good integrity check (e.g., cryptographic hash comparison).

16. Constraint Communication to the AI Service

This governance framework requires that governance constraints be communicated to the AI Service, enabling

it to self-constrain its recommendations. The specific mechanism for communicating constraints (system prompt, context injection, or other means) is an implementation choice.

16.1. Rationale

Communicating constraints to the AI Service serves two purposes:

- Efficiency: The AI Service can self-constrain its proposals, reducing the number of proposals that are generated, transmitted, parsed, and then rejected by the safety guardrails.

- Explainability: When the AI Service recommends escalation to a human operator, it can explain which governance constraints inform that recommendation, helping the human understand the situation.

Constraint communication does NOT replace programmatic enforcement. Regardless of what the AI Service is told, the agent MUST independently validate every proposed action against the governance rules before execution.

16.2. Constraint Categories

The following categories of constraints SHOULD be communicated to the AI Service in every prompt:

Action Scope: The list of allowed and blocked action types, derived from the Action Registry and Operator Allow/Block Lists.

Protected Targets: The patterns and names of protected resources (management interfaces, loopback interfaces, operator-defined resources).

Rate Limit Status: Information about applicable rate limits for remediation actions, enabling the AI Service to factor resource constraints into its recommendations.

Risk Ceiling: The maximum risk level permitted for autonomous execution.

Parameter Ranges: Valid ranges for action parameters (e.g., OSPF cost 1-65534).

Irreversible Action Rules: Any additional constraints on irreversible actions (e.g., critical severity required).

Absolute Prohibitions: A summary of the prohibitions from Section 14.

Uncertainty Guidance: An explicit instruction that when the AI Service is uncertain about the correct action, it SHOULD recommend human intervention rather than propose a potentially harmful action.

16.3. Device-Specific Context

When the AI Service is expected to recommend actions for a specific device, the prompt SHOULD include sufficient context about the target device to prevent recommendations that are syntactically valid in general but not applicable

to the specific platform or software version.

The specific mechanism for providing device context is an implementation choice and is outside the scope of this document. Implementers should ensure that any device context provided to the AI Service does not include credentials, secrets, or topology information beyond what is necessary for the specific recommendation request.

17. Implementation Considerations

17.1. Defense in Depth

Implementers SHOULD enforce governance constraints at multiple layers:

- AI prompt layer: Communicate constraints to reduce non-compliant proposals
- Response parsing layer: Validate AI response structure and extract only recognized action types
- Safety guardrail layer: Validate every proposed action against all governance rules before execution
- Execution layer: Validate parameters at the point of device interaction

No single layer should be considered sufficient. Defense in depth ensures that a failure in one layer (e.g., a parsing bug that accepts an unrecognized action) is caught by another layer (e.g., the safety guardrails reject actions not in the registry).

17.2. Fail-Safe Design

All governance checks SHOULD be designed to fail safe: in the event of an error in the governance logic itself, the default outcome SHOULD be to block the action rather than permit it.

For example:

- If rate limit data cannot be queried, assume the limit is exhausted (block)
- If protected target matching encounters an error, assume the target is protected (block)
- If the action registry cannot be consulted, assume the action is not registered (block)

17.3. Stateless Governance Checks

Governance checks SHOULD be stateless where possible, deriving their inputs from the persistent audit trail rather than in-memory state. This ensures that agent restarts do not reset rate limit counters or lose track of recent actions.

17.4. Post-Action Verification Accuracy

Implementers MUST ensure that post-action verification (Section 4.3) accurately reflects the current state of the target, rather than relying on cached or suppressed

detection results.

Verification that uses stale data may falsely indicate that an anomaly has been resolved when it persists, undermining the rollback safety mechanism.

17.5. Convergence-Sensitive Metric Set

Implementers SHOULD define a set of convergence-sensitive metrics appropriate for their device type and protocol support. A RECOMMENDED starting set includes:

- BGP peer session state changes
- IGP (OSPF/IS-IS) adjacency state changes
- IGP SPF computation count increases
- FIB/RIB update rate spikes
- BFD session state changes

The convergence detection threshold (number of simultaneous critical anomalies required to trigger suppression) SHOULD be configurable, with a RECOMMENDED default of 2.

17.6. AI Service Selection

This framework is AI-service-agnostic. Implementers MAY use any AI Service (LLM, expert system, or other reasoning engine) that can:

- Accept structured input (telemetry context, governance constraints)
- Return structured output (analysis, proposed actions, escalation flags)
- Be accessed via a stateless API (no persistent session required)

The governance framework does not depend on any specific AI Service’s capabilities, training data, or reasoning approach. The safety guarantees are provided by the governance layer, not by the AI Service.

18. Security Considerations

AI-mediated autonomous network management introduces several security considerations beyond those of traditional network management:

18.1. AI Service as Attack Surface

The AI Service API is a network-accessible endpoint. An attacker who can compromise the AI Service or intercept its responses could inject malicious remediation proposals.

Mitigation:

- All communication with the AI Service SHOULD use TLS with certificate validation
- The safety guardrails layer MUST validate all AI

proposals regardless of source, providing defense against compromised AI responses

- The absolute prohibitions (Section 14) cannot be overridden by any AI response

18.2. Credential Management

The agent requires credentials for device management interfaces (NETCONF, RESTCONF) and the AI Service API. These credentials MUST NOT be stored in plaintext configuration files.

RECOMMENDED: Store credentials in environment variables, hardware security modules, or platform-specific secret stores. The agent reads credentials at runtime only.

18.3. Prompt Injection

If the agent includes device-generated data (log messages, interface descriptions, SNMP community strings) in prompts to the AI Service, an attacker who can inject content into these data sources could influence the AI Service's reasoning.

Mitigation:

- The safety guardrails layer validates all AI outputs regardless of reasoning, so even a manipulated AI response is subject to governance checks

- Device-generated strings included in prompts SHOULD be sanitized or escaped

- The AI Service's role is advisory only; it cannot directly execute actions

18.4. Audit Trail Integrity

The audit trail is the primary accountability mechanism. If an attacker can modify or delete audit records, the governance framework's transparency guarantees are undermined.

RECOMMENDED: Store the audit trail in a write-ahead log database with integrity protections. Consider forwarding audit records to an external log aggregator for independent retention.

18.5. Exfiltration Risk

The agent necessarily transmits device state information to the AI Service for analysis. This creates a data exfiltration channel.

Mitigation:

- Limit the telemetry included in prompts to what is necessary for anomaly analysis

- Do not include full device credentials, topology databases, or customer data in prompts

- Use AI Service providers with appropriate data handling agreements

- The absolute prohibition against exfiltrating credentials and topology to unauthorized services (Section 14, item 9) provides a governance backstop

19. IANA Considerations

This document has no IANA actions.

20. References

20.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

20.2. Informative References

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, DOI 10.17487/RFC5424, March 2009, <https://www.rfc-editor.org/info/rfc5424>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/info/rfc8641>.

[OPENCONFIG] OpenConfig Working Group, "OpenConfig Models", <https://www.openconfig.net/>.

[GNMI] gNMI Specification, "gRPC Network Management Interface", <https://github.com/openconfig/gnmi>.

Appendix A. Degradation Level Reference Table

Level	Condition	Permitted Capabilities
0	All components healthy	Full autonomous operation

1	AI Service unavailable	Monitoring + detection + alerting (no remediation)
2	Notification channels down	Monitoring + detection + logging (no remediation)
3	Persistent storage unavailable	Logging only (no detection or remediation)
4	Telemetry collection failed	Safe shutdown

Table 1: Degradation Level Definitions

Appendix B. Rate Limit Reference Table

Limit	Default	Maximum
Remediation actions / hour	5	20
Actions / target / 24h	3	5
Irreversible / target / 24h	1	2
AI queries / hour	20	60
Anomaly cooldown (seconds)	300	none
Retry attempts / anomaly	3	5

Table 2: Rate Limit Defaults and Maximums

Appendix C. Absolute Prohibitions Checklist

The following checklist summarizes the absolute prohibitions from Section 14 for use in implementation verification:

- [ ] Configuration deletion blocked in all code paths
- [ ] Routing policy modification blocked
- [ ] BGP ASN modification blocked
- [ ] Bulk interface shutdown blocked
- [ ] Management plane config modification blocked
- [ ] Auth/authz modification blocked
- [ ] Self-modification of governance rules blocked
- [ ] Arbitrary CLI/shell execution blocked
- [ ] Credential/topology exfiltration blocked
- [ ] Operation without audit trail blocked

Acknowledgements

The concepts in this document are derived from operational experience deploying AI-mediated autonomous management systems on production network infrastructure. The author acknowledges the contributions of the broader network operations community in identifying the safety challenges that this framework addresses.

Author’s Address

Andrew Smith
Arrcus, Inc.
2077 Gateway Pl #400
San Jose, CA 95110
United States of America

Phone: +1-408-884-1965
Email: andy@arrcus.com

Nalin Pai
Arrcus, Inc.
2077 Gateway Pl #400
San Jose, CA 95110
United States of America

Phone: +1-408-884-1965
Email: nalin@arrcus.com