AiVRIC Policy & Rule Configuration Guide

This guide explains how to configure policies, rules, and control logic within AiVRIC Defense to tune detections, align to compliance frameworks, and support risk-based operations.

---

1. Overview of AiVRIC Policy Architecture

AiVRIC Defense evaluates posture using a layered policy model:

Policy Layers

- **Baseline Policies** – Core rules maintained by AiVRIC for cloud security, identity, logging, vulnerability posture, and compliance alignment.
- **Framework Policies** – Rules aligned to standards (PCI DSS, SOC 2, NIST 800-171, ISO 27001, CIS, etc.).
- **Custom Policies** – Customer-defined logic to detect configuration drift, required settings, or organization-specific governance requirements.
- **Exception Policies** – Scoped overrides to suppress or modify rules for assets, tags, or time-bound business requirements.

Evaluation Flow

1. Asset is scanned or ingested via API.
2. Baseline + framework rules are evaluated.
3. Custom rules are evaluated.
4. Exceptions are applied.
5. Resulting findings are scored and published to dashboards, exports, and integrations.

---

2. Baseline Policies

AiVRIC includes prebuilt best-practice policies for:

- Identity & Access Management
- Network exposure & segmentation

- Logging & telemetry completeness
- Cloud service configuration hardening
- Vulnerability status & patch gaps
- Encryption & key management

What You Can Modify

- Severity levels (Low / Medium / High / Critical)
- Tags and asset filters
- Alert routing (Slack, Teams, Jira, ServiceNow)
- Required remediation deadline windows

What You Cannot Modify

- Core rule logic that ensures minimum security baselines.

---

3. Framework Policies

Framework policies map AiVRIC findings to the control requirements for supported standards.

Supported frameworks include:

- **PCI DSS 4.0**
- **SOC 2 TSC**
- **ISO 27001:2022 Annex A**
- **NIST SP 800-171 / CMMC L2**
- **CIS Benchmarks**

How Framework Rules Work

- Framework policies automatically enable all relevant rules.
- Each rule carries metadata:
    - Control reference (e.g., ISO A.8.16)
    - Evidence type
    - Test method
    - Severity

You can enable, disable, or override rule behavior for specific frameworks.

---

4. Custom Rule Configuration

Custom rules allow organizations to extend AiVRIC's detection logic.

Custom Rule Types

- **Configuration Checks** – e.g., "S3 buckets tagged 'public' must have lifecycle rules enabled."
- **IAM Logic** – e.g., "All break-glass accounts must rotate keys every 24 hours."
- **Asset Group Policies** – e.g., "Production VMs must run only approved OS versions."
- **Log Coverage Rules** – e.g., "All critical resources must export logs to central SIEM."

Rule Builder Options

- Asset filters (tags, resource types, environments)
- Conditional operators
- Severity assignment
- Required evidence types
- Exceptions

Example Custom Rule

Goal: All production workloads must have encryption enabled.

IF asset.environment = "prod"
AND encryption = "disabled"
THEN severity = "High"
AND create finding "Encryption Required"

---

5. Exceptions & Temporary Overrides

Exceptions allow controlled deviation from policy.

Exception Attributes

- Scope: single resource, tag group, or subscription/account
- Duration: time-boxed (recommended) or permanent
- Business justification
- Approver
- Evidence attachment (ticket, approval doc)

Use Cases

- Legacy system awaiting replacement
- Temporary contractor access
- Migration-related exposure

Exceptions automatically expire unless renewed.

---

6. Severity, Scoring & Risk Weighting

AiVRIC calculates risk using:

- **Rule Severity** (Critical → Low)
- **Asset Criticality** (Business Impact)
- **Exposure Factors** (Public-facing, sensitive data, IAM access)

Customers can tune:

- Weight multipliers per environment
- Severity overrides
- Business impact ratings

---

7. Testing & Validating Rule Changes

Before publishing rule updates:

1. **Preview Impact** – See how many assets match the rule logic.
2. **Simulate Findings** – Test the rule against historical posture.
3. **Publish to Staging** – For enterprise editions with multi-workspace.
4. **Promote to Production** – After validation.

---

8. Recommended Policy Design Principles

- Keep rules simple and focused.
- Always time-box exceptions.
- Align custom rules with business risk.

- Review policies quarterly.
- Tie policy changes to architectural changes.

---

9. Ongoing Maintenance

AiVRIC supports versioned policy bundles.

Maintenance Cadence

- **Monthly:** Review failed controls
- **Quarterly:** Policy updates based on new services/features
- **Annually:** Framework alignment refresh

---

10. Appendix: Rule Builder Syntax Examples

Tag-Based Enforcement

IF tags.owner is NULL
THEN severity = "Medium"

Network Exposure Rule

IF public_exposure = TRUE
AND asset.type = "compute"
THEN create finding "Public Exposure Violation"

Identity Rule

IF iam.user.mfa = "disabled"
AND iam.user.role = "privileged"
THEN mark as Critical