CX-Catalyst - Best Practices

Operational best practices for knowledge base optimization, workflow tuning, and security hardening.

Table of Contents

Knowledge Base Optimization

Article Structure

Every KB article should follow a consistent format to maximize AI retrieval accuracy:

```
## Problem
[Clear, searchable description of the issue]

## Symptoms
[Observable indicators — error messages, behaviors, logs]

## Solution
[Numbered step-by-step resolution]

## Verification
[How to confirm the fix worked]

## Prerequisites
[Required access, tools, or conditions]

## Related Articles
[Links to related Confluence pages]
```

Writing Effective Titles

Titles are the single most important factor in vector search relevance.

Good titles: - "How to Reset Password for SSO Users" - "Resolving ERR_AUTH_001: Invalid OAuth Token" - "Configuring Webhook Retry Policies for Enterprise API"

Poor titles: - "Password Issue" - "Auth Error Fix" - "Webhook Help"

Tagging and Labels

Use Confluence labels consistently to improve categorization:

- **Product labels:** `web-portal`, `mobile-app`, `api`, `admin-console`
- **Issue type labels:** `authentication`, `billing`, `performance`, `configuration`

- **Severity labels:** `critical`, `high`, `medium`, `low`
- **Tier labels:** `enterprise`, `smb`, `small-business`

KB Coverage Targets

Aim for comprehensive coverage across all support categories:

Category	Target Articles	Priority
Authentication & Login	15+	High
Billing & Subscriptions	10+	High
API & Integration	15+	High
Configuration	10+	Medium
Performance & Scaling	10+	Medium
Security & Compliance	10+	Medium
Account Management	8+	Medium
Getting Started / Onboarding	5+	Low

Content Freshness

- Review and update articles **quarterly**
- Archive articles that reference deprecated features
- Add new edge cases as they are discovered during case reviews
- Monitor the `KB_GAP` comments from support agents and address them weekly

Embedding Quality

The system uses OpenAI `text-embedding-3-small` (1536 dimensions) for vector search:

- **Re-index after edits** — Run the KB Embedding Generator workflow after significant content changes
- **Avoid boilerplate** — Generic introductions dilute embedding quality; lead with the specific problem
- **One topic per article** — Articles covering multiple unrelated topics produce unfocused embeddings
- **Include error codes** — Exact error strings improve direct-match retrieval

---

Workflow Optimization

Confidence Threshold Tuning

Default thresholds balance automation with accuracy:

Threshold	Default	When to Adjust
Self-service (high)	85%	Lower to 80% if escalation volume is too high and AI accuracy is strong
Collaborative (medium)	60%	Raise to 65% if too many low-quality drafts reach the review queue
Escalation (low)	Below 60%	Lower floor if human agents are underloaded

Review threshold effectiveness monthly using the resolution-rate-by-confidence-tier SQL query in the Admin Guide.

Rate Limiting and API Cost Control

- **Batch API calls** — Use Loop Over Items with batch sizes of 5-10 for bulk operations
- **Add Wait nodes** — Insert 1-2 second delays between consecutive AI API calls to avoid rate limits
- **Pin test data** — During development, pin node outputs to avoid consuming API tokens
- **Monitor token usage** — Use the Grafana dashboard Token Usage panel to track costs by workflow
- **Right-size models** — Use Claude Sonnet for classification and solution generation; reserve larger models for complex analysis

Workflow Execution Efficiency

- **Keep workflows under 20 nodes** — Break larger flows into sub-workflows
- **Use sub-workflows for shared logic** — Deduplicates KB search, classification, and notification patterns
- **Set execution timeouts** — 3600s (1 hour) for standard workflows; adjust for Workflow 5 which runs longer
- **Enable error workflows** — Every production workflow should have an error handler assigned
- **Use expression mode carefully** — Complex expressions in node parameters are harder to debug than Code nodes

Review Queue Management

- **Target 2-hour SLA** — Configure timeout alerts at 90 minutes
- **Balance queue load** — If queue depth exceeds 50 pending cases, consider lowering the self-service threshold temporarily
- **Track edit patterns** — High edit rates on a specific category indicate a KB gap or prompt tuning opportunity
- **Rotate reviewers** — Distribute reviews across the team to build shared expertise

---

Security Best Practices

Credential Management

- **Never hardcode secrets** — All API keys, tokens, and passwords must be stored in the n8n credentials store or environment variables
- **Rotate keys quarterly** — Follow the rotation procedure in the Admin Guide (Section: API Key Rotation)
- **Use separate credentials per environment** — Development and production should use different API keys
- **Audit credential access** — Review who has access to n8n credentials monthly

Webhook Security

- **Use HTTPS** — All webhook endpoints must use TLS encryption
- **Implement path tokens** — Add secret tokens to webhook URLs for basic authentication
- **Validate request headers** — Use `X-API-Key` header validation for programmatic callers
- **IP allowlisting** — Restrict webhook access to known source IPs where possible
- **Rate limit at the load balancer** — Prevent abuse before requests reach n8n

Data Protection

- **Minimize data in logs** — Avoid logging customer PII in workflow execution outputs
- **Encrypt at rest** — Ensure PostgreSQL and Supabase use encrypted storage
- **Encrypt in transit** — All API communication must use TLS 1.2+
- **Limit database access** — Use role-based database users (read-only for reporting, read-write for n8n service)
- **Audit database access** — Enable PostgreSQL query logging for the service account

AI-Specific Security

- **Review AI outputs** — Medium-confidence cases go through human review before reaching customers
- **Monitor for hallucinations** — Track cases where AI solutions are rejected; high rejection rates indicate prompt or KB issues
- **Sanitize inputs** — Validate and sanitize customer descriptions before passing to AI prompts to prevent prompt injection
- **Data residency** — Understand where Anthropic and OpenAI process your data; review their data handling policies
- **No sensitive data in prompts** — Avoid including customer credentials, payment information, or PII in AI prompts

Network Security

- **WAF protection** — Deploy a web application firewall in front of public-facing webhook endpoints
- **Segment networks** — Keep the n8n instance, database, and monitoring on a private network where possible

- **Monitor for anomalies** — Set up alerts for unusual webhook traffic patterns (volume spikes, unusual source IPs)
