

One MCP Server, Many Agents

The Identity Gap OAuth Doesn't Close



Mohit Gurnani

Member of Technical Staff 4 · Nutanix

NUTANIX

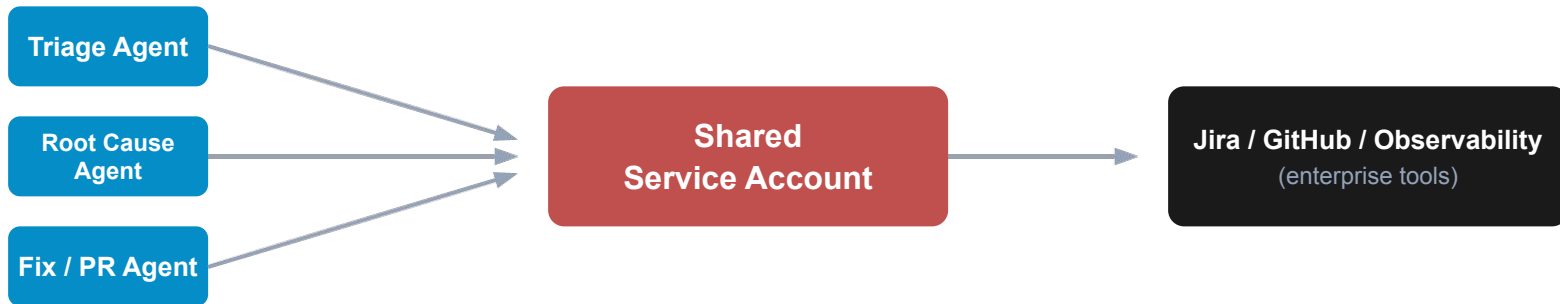
Overview & Topics

- 01 The Problem
- 02 Concepts at a Glance
- 03 The Gap
- 04 Decision Framework

- 05 Envoy AI Gateway
- 06 Security Boundary

THE PROBLEM

One shared identity, many agents



THE SHARED SHORTCUT

Every agent authenticates as the same service account.

READ BEYOND USER PERMISSIONS

The bot can read data the requesting engineer cannot access.

IDENTITY COLLAPSE

Downstream logs see the bot, not the initiating user. No per-user authorization at the downstream service.

Three core terms in plain language

OAuth

Delegated Access

An application obtains access on behalf of a user, subject to the authorization flow and scopes.

Client Credentials

Machine-to-Machine

A service proves its own identity. No human involved.

Header / PAT Forwarding

Credential Pass-Through

An existing credential is passed to the downstream tool, which validates it and enforces its permissions.

Two Authentication Patterns. One Missing Identity.

1

OAuth 2.1

Interactive, browser-based consent

- Requires active user present in browser session
- Relies on real-time human authorization

✗ Fails at 2am

Headless agent has no human at the browser to grant interactive consent.

2

Client Credentials

Headless, fixed scope — agent's identity

- Pure machine-to-machine authentication
- Executes autonomously using shared service bot credentials

✗ Fails on Accountability

Fixed 'sre-bot' identity masks human user when Alex takes an on-call action.

Neither preserves a specific user's identity for a headless agent.

When Alex is on-call at 2am, actions collapse into 'sre-bot' — leaving zero audit trail of who was responsible.

Four patterns, one decision

Pattern	Headless-safe	Per-user RBAC?	Additional Infra	Best fit
Interactive OAuth 2.1	No	Yes	High	Interactive IDE clients
Client Credentials	Yes	No (machine only)	Low	Standalone automation
RFC 8693 Exchange	Yes	Yes	Medium	Cross-IdP / cross-domain
Header / PAT Forwarding	Yes	Yes	None	Headless agents inheriting identity

Header forwarding is the only pattern with zero new gateway infrastructure — right when the user's own credential already works downstream.

Zero-Infra Identity Propagation for Agent Workflows



1. Headless Agents

- User-scoped PAT stored in Vault
- Synced by External Secrets
- Mounted as a file
- Audit logs say **Alex**, not **sre-bot**.



2. Local IDE / CLI

- Your PAT, forwarded as-is.
- Preserves User Identity & RBAC.



3. Legacy / Native MCPs

- GitHub & Atlassian explicitly prefer PATs over OAuth for write attribution.
- Many MCP servers (ClickHouse, Gerrit) don't support OAuth at all.

Before: sre-bot → After: Alex | PAT forwarding closes the identity gap. Major platforms (GitHub, Atlassian, Block/Goose) prefer header forwarding for true write attribution.

Atlassian Gateway Configuration Example

```
mcp.json

"atlassian": {
  "url": "<MCP Hosted URL>",
  "headers": {
    "X-Atlassian-Jira-Personal-Token" : "<Your token>",
    "X-Atlassian-Confluence-Personal-Token" : "<Your token>"
  },
  "type": "http"
}
```

KEY TAKEAWAY

One header per user. The gateway forwards it. Jira sees Alex.



Per-User Header Ingestion

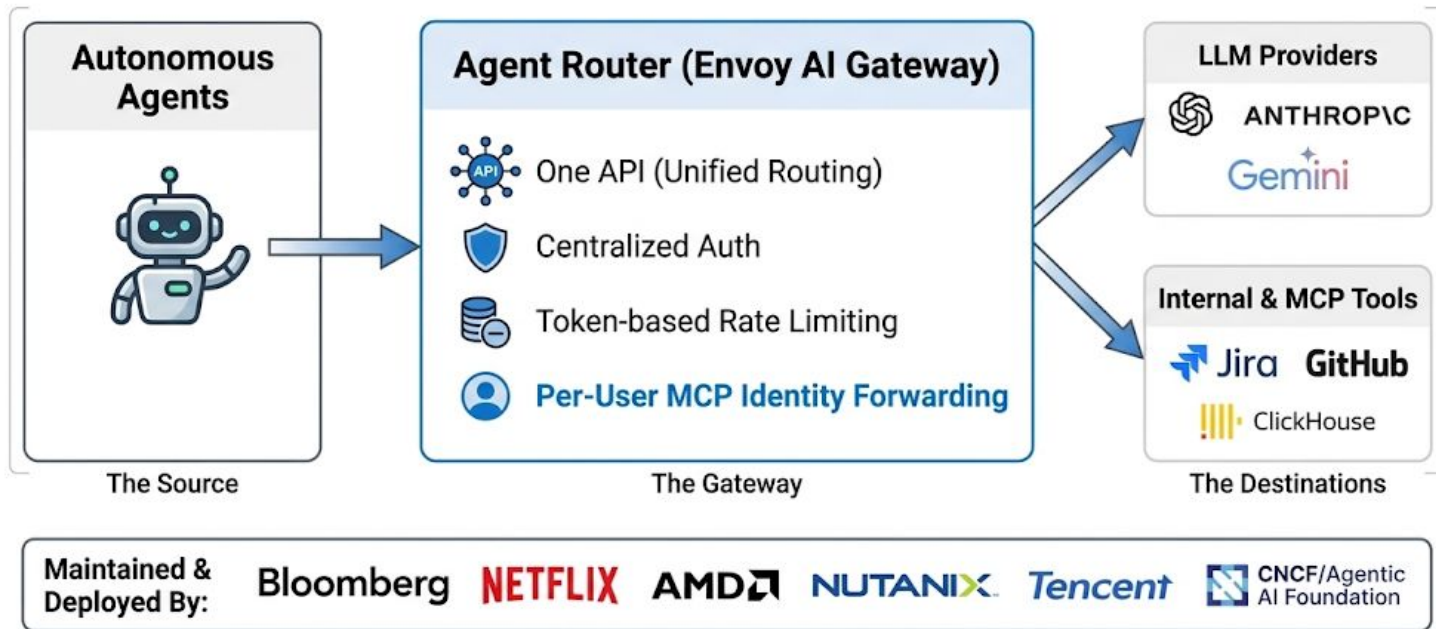
Configures dedicated Personal Access Tokens for both Jira and Confluence endpoints.



Identity Transparency

Transparently forwards credentials so downstream services log actions under individual user identities.

Production-grade MCP & LLM governance



ANNOUNCEMENTS

2026-09-09

Envoy AI Gateway is now agentrouter, an Agentic AI Foundation project

Same code. Same maintainers. New home. Joined the
Agentic AI Foundation on 10 September.

theagentrouter.ai



Scoped tokens are safe. Broad tokens are not.

Scoped Tokens (PAT) — Low Risk

Single-Purpose & Restricted

Explicitly restricted to specific actions (e.g., read-only access on a single Jira project).

Isolated Blast Radius

If leaked, stolen, or misused, damage is strictly confined to that single project.

Broad Login Tokens (OIDC) — High Risk

Confused Deputy Vulnerability

Forwarding general identity tokens internally creates severe impersonation risks.

Unchecked Audience Scope

If receiving services fail to validate intended audience, an internal tool accepts tokens to access sensitive backends like **Payroll**.

CORE SECURITY RULE

Forward scoped credentials, and only to the backend they were issued for.

"The identity already existed. It just wasn't flowing through."

- **No Huge Auth System Needed**

You don't need to construct custom authentication layers for most architecture cases.

- **One Field Config**

Set `forwardHeaders: on` directly on your MCP route config with zero infra overhead.

- **Bridge the Gap**

Simply allow the credential that already exists to actually reach the target tool.

Blog Guide



Scan to read full article

theagentrouter.ai/blog/multi-user-mcp-header-forwarding

Thank You

Let's Connect



Mohit Gurnani

MTS 4 @ Nutanix, Inc.

LinkedIn

linkedin.com/in/mohitgurnani94



Connect

Scan to connect



One MCP Server, Many Agents. The Identity Gap (OAuth) Doesn't Close.

Feedback

Scan for feedback

