Darwinium

SOLUTION BRIEF

MCP Observability & Protection

Bring your fastest-growing attack surface under security control — observe, authenticate, and govern every MCP tool call at the edge.

FOR SECURITY & RISK LEADERS

Enable AI Commerce. Stop AI Fraud.

Released in Darwinium 1.5.24

darwinium.com
2026

THE BLIND SPOT

A new attack surface your existing controls can't see

AI agents now act autonomously against your services. They discover, transact, and operate on a user's behalf by calling "tools" over the Model Context Protocol (MCP). The emerging commerce standards are converging on it: the Universal Commerce Protocol (UCP), backed by Google, Shopify and a roster of major retailers and networks, supports REST, JSON-RPC and MCP transports natively, as do related efforts such as ACP and the AP2 agent-payments protocol.

For a security team, an MCP endpoint is a **new, internet-facing attack surface** — and it sits *outside* the controls you have spent years building. Your WAF, bot-management and identity stack were built to inspect browser page loads and REST APIs. They do not parse JSON-RPC tool calls, cannot see which tools are being invoked with which arguments, and have no concept of the autonomous, **non-human identities** now driving that traffic. The result is a governance gap, and the oldest problem in security: **you cannot protect what you cannot see.**

That blind spot is being actively probed. The same tool calls that let a *trusted* agent run a product search or build a cart let an attacker do the same at machine speed:

- **Automated credential and payment-card testing** against agentic checkout endpoints.
- **Data exfiltration and scraping** — systematic theft of catalogue, pricing and content via search tools.
- **Business-logic abuse** — inventory hoarding, coupon stacking, flash-sale exploitation.
- **Resource exhaustion and cost amplification** — API-driven denial-of-service and infrastructure cost attacks.
- **Hijacked or prompt-injected agents** — a legitimate agent commandeered, its session loaded with stolen credentials or steered off-mandate.

And the defender's classic challenge is amplified: **legitimate and malicious automation are indistinguishable at the packet level.** The old model — *block the bot, authenticate the human* — collapses when non-human identities are simultaneously your customers' assistants and your attackers' tooling, and when cloud-executed agents concentrate thousands of "users" behind a handful of subnets.

97%

of organizations experienced increased **AI-assisted fraud attacks** in the last 12 months — yet a third still **block agent traffic by default**, and nearly two-thirds **can't detect it** across the full customer journey.

DARWINIUM RESEARCH · FEBRUARY 2026

The control question has shifted.

THE OLD QUESTION

"Is this a bot?"

THE QUESTION THAT MATTERS NOW

"Is this non-human identity authorized — and is this tool call consistent with legitimate intent, right now?"

THREATS ADDRESSED

What you're actually up against

The abuse arriving over MCP maps cleanly to the **OWASP API Security Top 10** and the **OWASP Top 10 for LLM Applications** — so it slots straight into your existing risk register. Here is what it looks like, and how Darwinium responds. *(The agentic / LLM categories are newer and still evolving.)*

Credential & payment-card testing

API6 · API2

A burst of cart and payment tool calls, near-zero time between steps, a path no real user takes.

→ **Challenge or block at the edge, before the request reaches your payment backend.**

Data exfiltration & scraping

API6

Systematic enumeration of product IDs at uniform cadence, no engagement signals, tool calls without any normal session flow.

→ **Throttle or block; protect your proprietary data and your infrastructure spend.**

Business-logic abuse

API6 · SCALPING / HOARDING

Rapid add-to-cart loops on limited stock, velocity far in the tail of normal.

→ **Step up or rate-limit, while letting legitimate agent activity through.**

Session hijack / prompt injection

LLM01 · EXCESSIVE AGENCY

The agent carries a *valid* signature — identity checks pass — but its journey diverges from the declared mandate and its actions are inconsistent with intent.

→ **Contain or block anyway. Identity alone never proved the session wasn't compromised; behavior does.**

Internal & east-west MCP services

API9 · INVENTORY MGMT

The same controls apply to internal tool servers and partner integrations that speak JSON-RPC: observe the tool calls, baseline normal, and alert on drift.

→ **Extend governance to surfaces you didn't previously instrument.**

THE CAPABILITY

01 What Darwinium MCP Observability & Protection does

Released in Darwinium 1.5.24, this capability brings MCP traffic under the same edge-native security control plane that already governs your web, API and mobile traffic — with full visibility, non-human identity authentication, behavioral threat detection, and real-time enforcement. It works on any web service that speaks JSON-RPC; agentic commerce is simply the sharpest example.

1 A forensic record for every agent session

It's what the punters came to see. Individual RPC calls tell you little on their own; the **sequence is the evidence**. Darwinium parses JSON-RPC at the perimeter edge and sees exactly **which tool the agent is calling** — then stitches the agent's ordered tool calls and the eventual human checkout into a **single journey**, one journey ID, with payloads and geolocation visible on every call. You choose which tool calls to inspect, the same way you already govern API traffic, with **no backend code changes**.

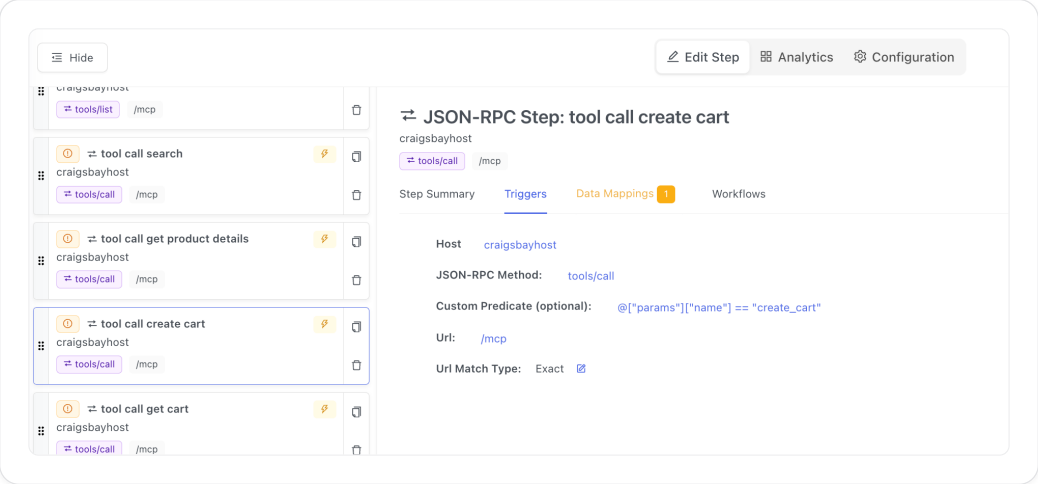

FIG 1 Select the exact tool call to observe — here the `create_cart` tool on the `tools/call` method, configured entirely from the portal.

Darwinium then **maps the parameters the agent is passing** — any field a tool call carries becomes an attribute your policies and downstream analytics can use. For your SOC and incident responders the result is a complete, attributable audit trail: the agent's prompt, the tools it called and in what order, what it transacted, and where each call originated, through to the final action. **Forensic clarity** for investigations; non-repudiable evidence for dispute and chargeback defence.

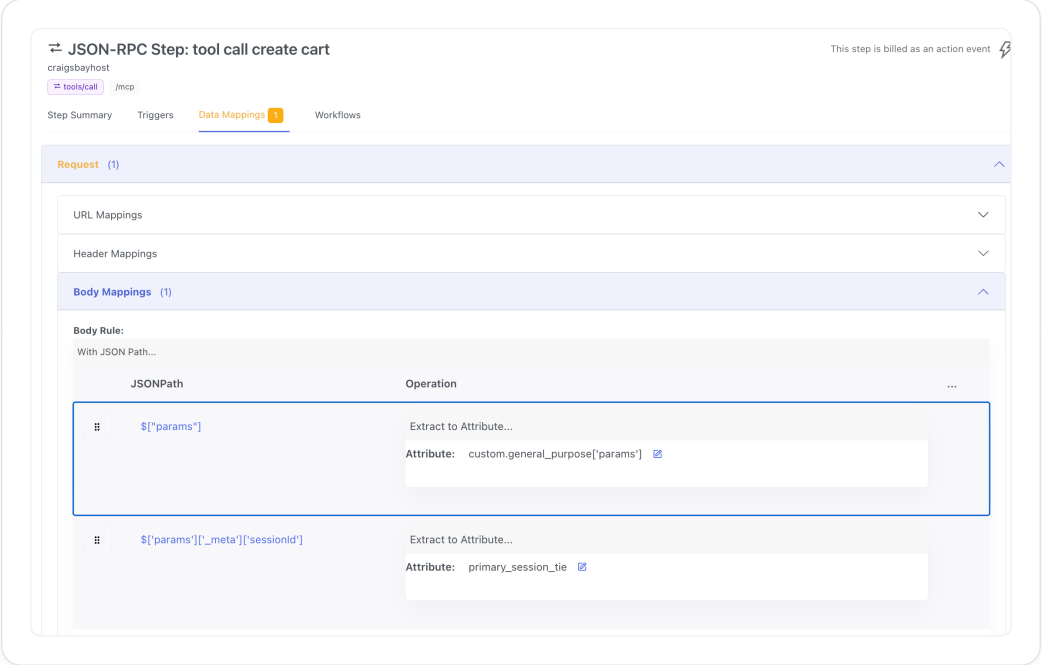

FIG 2 Map the parameters the agent is passing — each field becomes an attribute Darwinium can use in risk decisioning and remediation.

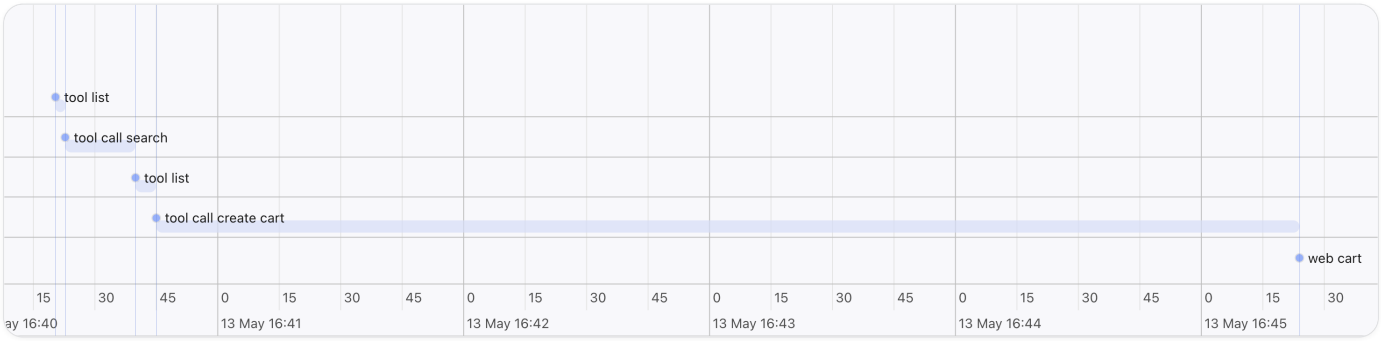

FIG 3 One session, one timeline: tool list → tool-call search → tool-call create cart → human web cart.



FIG 4 Payloads are fully visible — line-item id, quantity, and cart totals parsed straight from the JSON-RPC call.

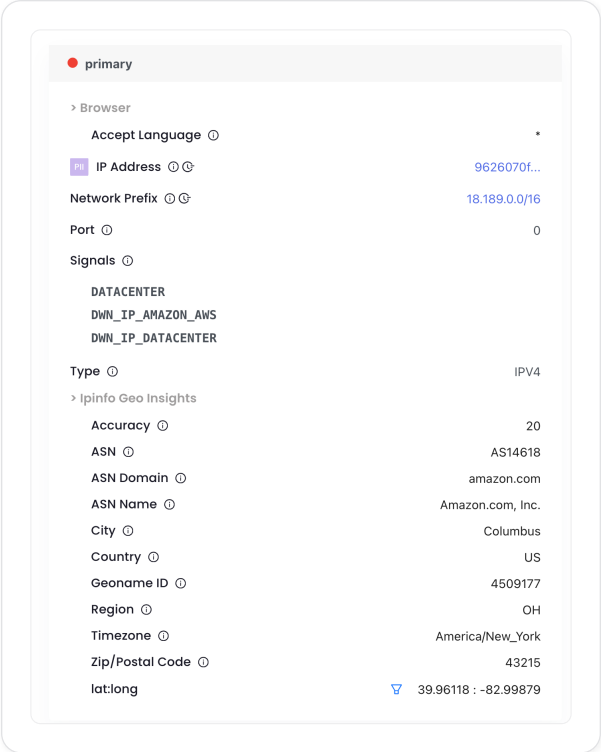

FIG 5 Where each call originated — datacenter / AWS detection and per-call geolocation for attribution.

2 Authenticate non-human identities at the edge

Emerging standards such as **web-bot-auth** let a legitimate agent cryptographically sign its requests so you can verify *who* it is. Darwinium validates these **HTTP message signatures at the perimeter edge** — surfacing whether a signature is present and, as the standard matures, whether it validated — and lets you make that result a policy signal: **conditionally permitting specific tool calls based on the identity's risk profile.** This is zero-trust extended to the agents acting on your estate.

Critically, this is **one primitive that covers many schemes** — web-bot-auth, Visa TAP, Mastercard Agent Pay and others all use the same signature pattern — so a single integration future-proofs you against the protocol your partners mandate next. And you adopt it at your own pace, through four operating modes.

Validate

Verify the signature at the edge and surface every claim as a signal; your backend never has to parse it.

Monitor

Run passively: see which traffic is signed, by whom, and what it's authorizing. The fastest answer to "what does our agent traffic actually look like today?"

Enforce

Require a valid signature on sensitive tool calls or endpoints; step up or block at the edge before the request reaches your backend.

Translate

Inject the verified claims as plain HTTP headers your existing security and order systems already know how to read.

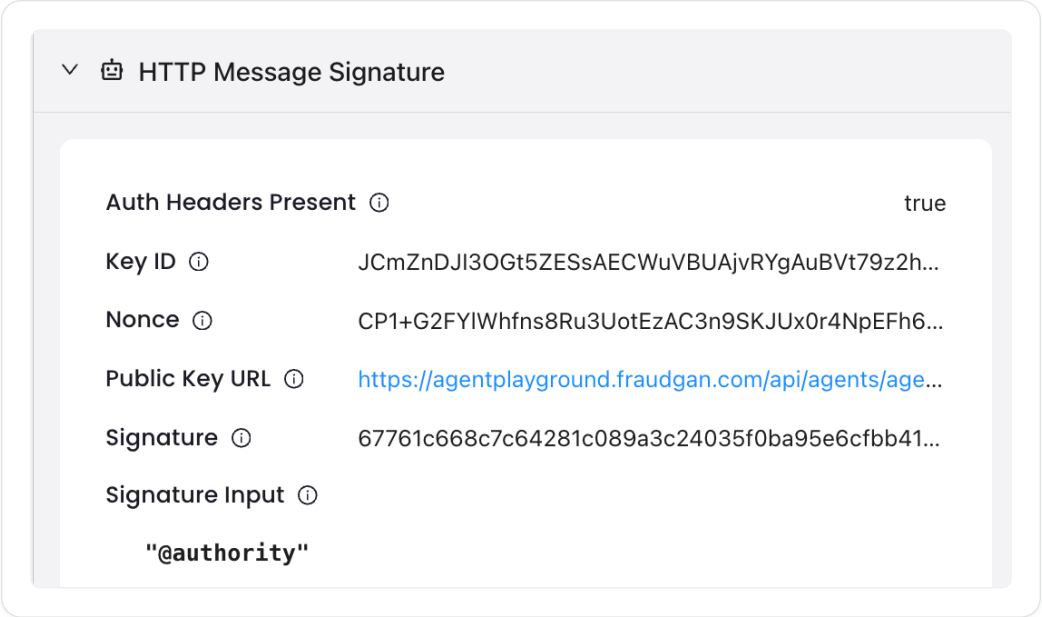

FIG 6 web-bot-auth in action: the agent's HTTP message signature, key and public-key URL, surfaced and validated at the edge.

3 Detect malicious intent without signatures or training

Authentication tells you *who*; it cannot tell you whether a verified identity has been hijacked or is acting maliciously. Darwinium's **Journey Transition Probability** model closes that gap. It has run in production for years as a universal behavioral signal, and because it scores every event, it scores **every MCP tool call** — no attack signatures to maintain, no labelled training data required.

It learns the *shape* of legitimate traffic and scores every new tool call against it, producing four signals your analysts can rule on individually or together:

SIGNAL	WHAT IT CATCHES
Per-step probability	Skipped steps and unfamiliar transitions (e.g. straight to checkout, no browse).
Path probability	Whole sequences no legitimate actor takes; coordinated abuse patterns.
First-passage time	Unnatural, programmatic speed — or suspicious hesitation.
Transition-rate change	A shift in your *normal* — novel attack TTPs *before* any labelled incident.



FIG 7 Per-journey transition probabilities: each step scored against normal, with the low-probability withdrawal flagged in-line.

The model is layered — a **global** baseline across all Darwinium customers, your own **per-organization** normal, and a **per-journey** model for each flow — so you get a useful signal on **day one** and it **hardens automatically** as your own traffic accumulates. This is how Darwinium flags hostile MCP traffic out of the box, including threats no signature has been written for yet.

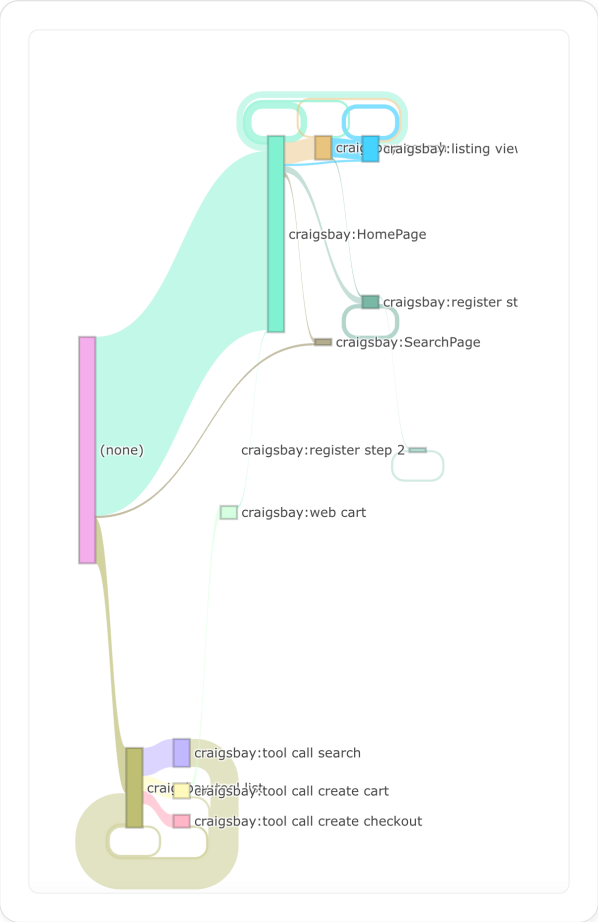

FIG 8 The wider journey / transition flow across web pages and MCP tool calls — the structure the model scores every call against.

CAPABILITY SPOTLIGHT

Risk-based access for non-human identities

This is zero-trust made concrete for agents. Because Darwinium scores both identity *and* behavior in real time, you can apply **adaptive, risk-based enforcement** to the checkout (or any sensitive action) — containing risk without blanket-blocking the automation your business now depends on.

TRUSTED · AUTHENTICATED

Frictionless prompt-to-payment

A valid signature, a journey consistent with legitimate intent, low risk — granted an end-to-end path that completes with no human in the loop.

LOWER-TRUST · UNVERIFIED

Contained in a web checkout

Routed into a web-based flow where Darwinium applies its full profiling stack — device recognition, behavioral biometrics, network analysis — and can require a **human-present step** before any money moves.

Same destination, two paths, chosen by risk in the moment. **Never trust identity alone; verify behavior continuously; contain when risk rises.** It lets security say *yes* to AI-driven automation while keeping a hard control point exactly where the risk concentrates.

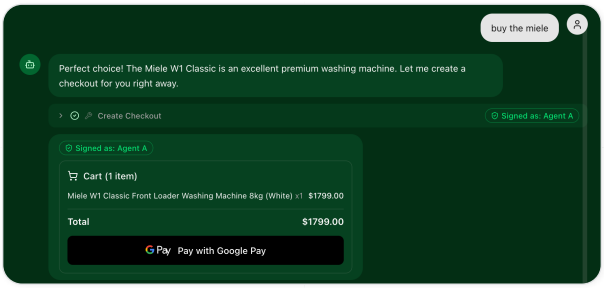

FIG 9 Trusted, signed agent (Agent A) → frictionless prompt-to-payment, completed inline.

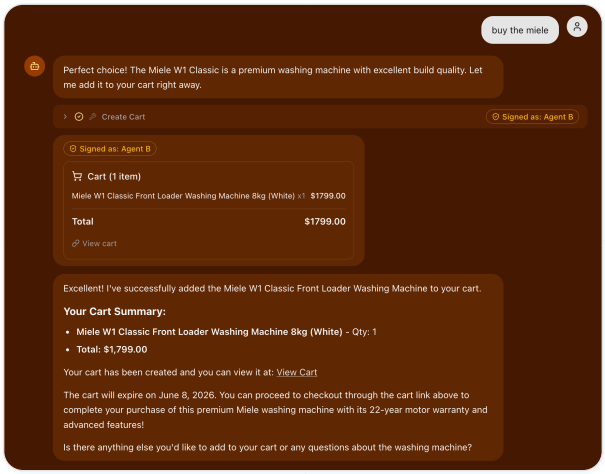

FIG 10 Lower-trust agent (Agent B) → contained: the purchase is pushed into the web checkout flow, where full profiling and a human-present step apply.

UNDERSTAND & PROTECT

One control plane for every agent protocol

MCP is the front door for agents — and, for a security team, a new internet-facing attack surface. Your WAF, bot-management and identity stack were built for browser page loads and REST APIs; they don't parse JSON-RPC tool calls. Darwinium does. By treating JSON-RPC as a **first-class payload type**, it eliminates the MCP blind spot and brings MCP — together with the fast-emerging stack of agent and payment protocols around it — under **one edge-native control plane**. One integration turns an unmonitored surface into instrumented telemetry, with understanding and enforcement across every protocol below.

- **Model Context Protocol (MCP)**

  The open, JSON-RPC standard agents use to discover and call your tools. Darwinium sees **which tool is invoked with which parameters**, records the full session, and scores every call — the visibility everything else builds on.

- **web-bot-auth** · Visa TAP · Mastercard Agent Pay

  Lets a legitimate agent cryptographically sign its requests so you can verify *who* it is. Darwinium validates these HTTP message signatures at the edge and makes the result a policy signal — **one primitive covering the shared signature pattern** behind all three schemes.

- **AP2 — Agent Payments Protocol**

  Delegated mandates and tokens that prove a payment is authorized. Darwinium adds the question they leave open: **is the session coerced or compromised before money moves?**

- **UCP / ACP — Universal Commerce Protocol**

  Backed by Google, Shopify and major retailers and networks; defines how agents transact over REST, JSON-RPC and MCP. Darwinium governs abuse **end-to-end across whichever transport** your partners mandate next.

Each standard answers one question. Darwinium answers the one they all leave open — and it speaks all of them.

The foundation, not another point tool

- **Consolidation, not another point tool**

A single Darwinium deployment at your CDN / edge governs **web, API, mobile *and* MCP** from one policy engine your team already operates. New protocols light up as new signals — reducing tool sprawl, not adding to it.

- **Defense-in-depth that complements your stack**

Darwinium sits *above* your CDN, WAF and bot-management layer. They answer "is this a known agent?"; Darwinium adds behavior, intent and enforcement on top — without forcing a rip-and-replace.

- **Edge-native, real-time, governed**

Decisions are made at the perimeter in milliseconds — on the CDN you already run (AWS CloudFront, Cloudflare or Akamai). Remediation is risk-based and new endpoints are brought under control from the portal, with a full audit trail for your SOC, risk and disputes teams.

`Accept` `Enrich` `Challenge` `Block`

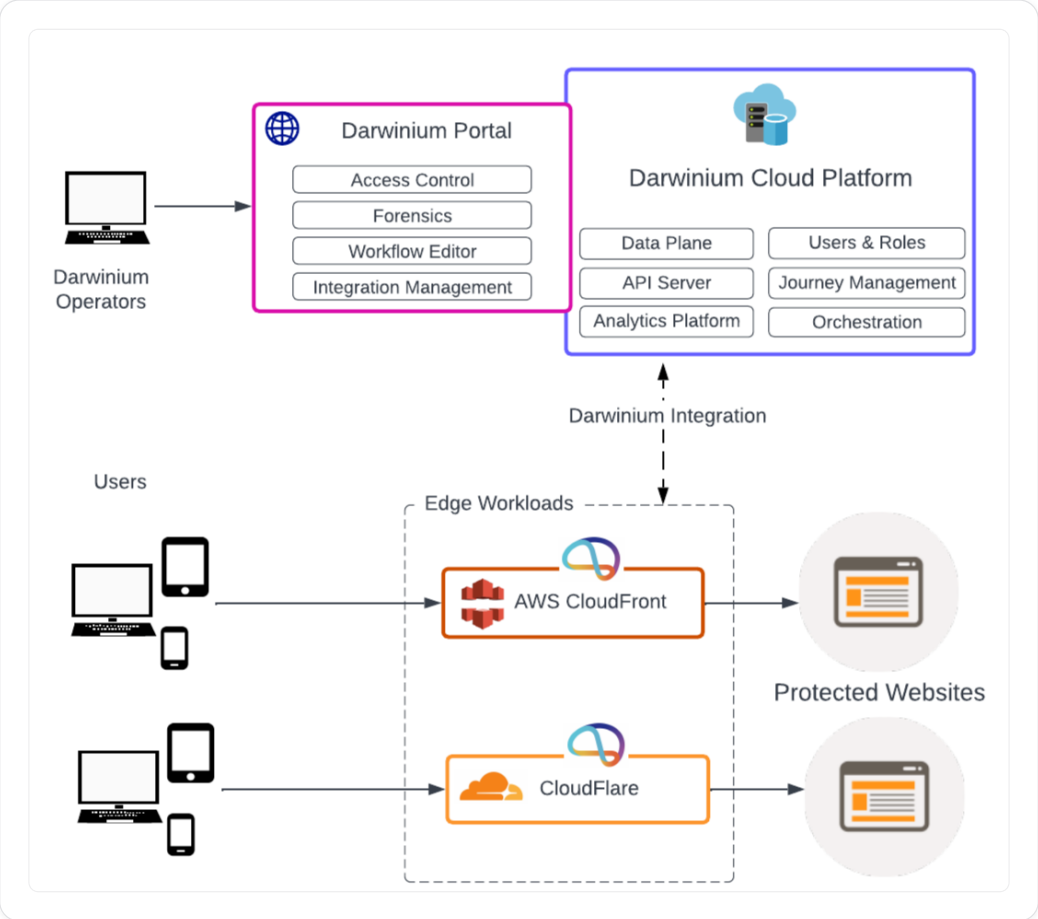

FIG 11 One deployment at the CDN edge — AWS CloudFront, Cloudflare or Akamai — governs web, API, mobile and MCP traffic, feeding a single cloud platform for analytics, journey management and orchestration.

Darwinium

ABOUT DARWINIUM

Continuous customer protection

Darwinium unites digital security with fraud prevention across every customer journey, protecting your end users from fraud, scams and online abuse. With full visibility of trust and risk across your entire digital estate, businesses can accurately separate good and bad behavior — removing the reliance on “point-in-time” API-based solutions that are vulnerable to exploitation.

Combine continuous security analytics with customer journey orchestration to make better risk decisions, and take real-time action on evolving threats, via one simple integration — so security can confidently enable AI-driven automation rather than block it.

Enable AI Commerce. Stop AI Fraud.

Darwinium — Continuous Customer Protection.