Cloudflare AI Security

Three Pillars of Protection for the AI Era

Pillar 1
End-User Protection
Gateway + DLP + WARP

Pillar 2
App & API Security
AI Gateway

Pillar 3
Agentic & MCP Security
MCP Portal + WAF

Three Distinct AI Threat Models

Pillar 1
End-User Protection

Who: Employees using ChatGPT, Claude, Copilot, Gemini via browser

- Corporate data leakage to AI providers
- Shadow AI without IT visibility
- PII and secrets in prompts
- Compliance violations

Solution: Cloudflare One
Gateway + WARP + DLP + CASB

Pillar 2
App & API Security

Who: Your applications calling LLM APIs programmatically

- Prompt injection via API
- Runaway costs from agent loops
- No visibility into LLM usage
- Single provider dependency

Solution: AI Gateway
Observability + rate limiting + DLP

Pillar 3
Agentic & MCP Security

Who: AI agents using MCP to access tools, APIs, and data

- Uncontrolled tool access
- Prompt injection via tool output
- Shadow MCP servers
- Inbound attacks on public MCP servers

Solution: MCP Portal + WAF
Access + AI Security for Apps

Pillar 1

End-User AI Protection

Cloudflare One — Gateway + WARP + DLP + CASB

The Shadow AI Problem

What IT Can't See

- Employees pasting source code into ChatGPT
- Customer PII shared with AI assistants
- API keys and secrets in prompts
- Strategic documents uploaded for summarization
- Proprietary algorithms explained to AI

Real Risk Scenarios

Data Breach: Engineer pastes production database credentials while debugging with AI

IP Theft: Product manager uploads unreleased roadmap for "quick summary"

Compliance: HR uploads employee records to AI for analysis

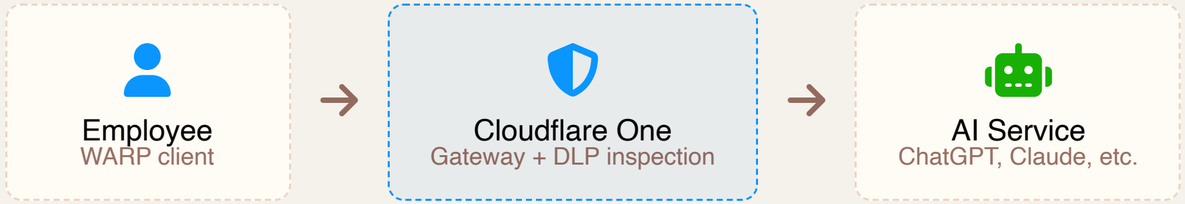
Legal: Attorney shares privileged client communications

The Scale

- **42+** AI applications tracked in Gateway
- **75%** of employees use AI tools at work*
- **60%** do so without IT approval*

*Industry surveys 2024

Pillar 1: Solution Architecture



WARP routes employee traffic through Cloudflare's network — DLP scans the HTTP body before it reaches the AI provider

Access

- SSO/MFA for AI tools
- Group-based policies
- Device posture checks

Gateway (SWG)

- 42+ AI apps in library
- Allow/block AI services
- Full HTTP inspection

DLP

- 700+ built-in detectors
- Scan prompt body for PII
- Block, warn, or log

Browser Isolation

- AI UI renders on Cloudflare edge
- Block copy/paste of data
- Disable file uploads

Zero Trust AI Controls

Access

- Require SSO/MFA for AI tools
- Group-based policies (only R&D can use Copilot)
- Device posture checks before AI access
- Session duration limits

Gateway (SWG)

- 42+ AI apps in application library
- Allow/Block specific AI services
- DLP scanning on prompts & uploads
- Full conversation logging

Browser Isolation

- AI interfaces render on Cloudflare edge
- Block copy/paste of sensitive data
- Disable file uploads to AI chats
- Prevent downloads of AI outputs

Data Loss Prevention

- 700+ built-in detectors (SSN, CC, etc.)
- Custom patterns for proprietary data
- Exact Data Match for sensitive lists
- Action: Block, warn, or log

Pillar 1: Customer Benefits

Complete Visibility

- See all AI tools employees are using
- Full conversation logging with user attribution
- Shadow AI discovery across the org
- Real-time dashboards and alerts

Compliance Ready

- PII never reaches AI providers
- Audit trails for every interaction
- GDPR, HIPAA, SOC2 alignment
- Data residency controls

Granular Control

- Per-user, per-group, per-app policies
- Block, warn, or log based on context
- Custom topic filtering
- Flexible exception handling

Enable Innovation Safely

- Say "yes" to AI with guardrails
- Empower employees without risk
- No productivity sacrifice for security
- Unified policy across all AI tools

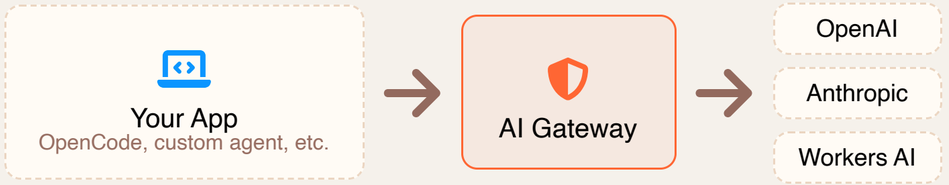
Pillar 2

App & API AI Security

AI Gateway — for developers and applications calling LLM APIs

What is AI Gateway?

Key distinction: AI Gateway is for developers and applications making programmatic LLM API calls — not for end users browsing to Claude.ai or ChatGPT. Those users are covered by Pillar 1 (Gateway + WARP + DLP).



Who uses this?

- Developers building AI-powered applications
- Applications like OpenCode routing to an LLM
- Internal AI assistants your company has built
- AI agents making repeated LLM calls

20+ Providers
Single endpoint, any model

Caching
Up to 90% cost reduction

Rate Limiting
Prevent runaway costs

Observability
Tokens, latency, cost

The App & Agent Security Challenge

The New Application Stack

Modern apps increasingly rely on LLM APIs:

- Customer-facing chatbots
- Internal copilots and assistants
- Automated content generation
- AI agents acting autonomously
- Agentic coding tools (OpenCode, Cursor, etc.)

Cost Reality

A single runaway agent loop can burn through **$10,000+** in API costs in minutes

OWASP Top LLM Threats

LLM01: Prompt Injection
Malicious inputs hijack model behavior

LLM02: Insecure Output
Model responses executed as code

LLM05: Supply Chain
Compromised models or training data

LLM06: Sensitive Info Disclosure
Models leak training data or PII

AI Gateway: DLP for Prompts & Responses

Scan both directions — protect data going to LLMs AND validate what comes back. No TLS decryption required.

→ Prompt Scanning (Outbound to LLM)

- Detect PII before it reaches the LLM
- Block API keys, passwords, secrets
- Flag proprietary code patterns
- Custom regex for business data

Actions: Block, redact, warn, log

← Response Scanning (Inbound from LLM)

- Catch hallucinated PII in responses
- Detect prompt leakage attacks
- Block toxic or harmful content
- Validate response format/structure

Actions: Block, sanitize, flag for review

Traditional DLP cannot inspect API traffic to AI providers. AI Gateway provides native visibility without SSL inspection complexity — because your app routes through it by design.

AI Gateway: Dynamic Routing & Observability

Routing Capabilities

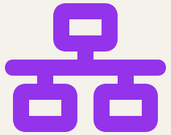
- **Fallbacks:** Auto-failover when provider is down
- **A/B Testing:** Split traffic between models
- **Conditionals:** Route by user, content, metadata
- **Budget Limits:** Cap spend per model/user
- **Rate Limits:** Requests per minute controls

Example Flow:
Try GPT-4 → If rate limited → Fallback to Claude → If budget exceeded → Use Workers AI

Observability

Every request logged with full context:

- Latency (TTFT, total)
- Token counts
- Cost per request
- Success/error rates
- Cache hit ratio
- Guardrail triggers



Pillar 3

Agentic & MCP Security

MCP Server Portal + AI Gateway + AI Security for Apps

What is Model Context Protocol (MCP)?

MCP is an open standard that lets AI models move beyond conversation — they can take actions by connecting to external systems.

Two Components

MCP Client — the AI-facing side
Claude Desktop, Cursor, OpenCode, ChatGPT. This is the LLM interface the user interacts with.

MCP Server — the tool-facing side
Exposes tools the AI can call: Jira, Slack, GitHub, internal databases, APIs. Each server wraps a specific resource.

What are "Tools"?

Actions the AI can invoke — search Jira, query a database, send a Slack message, read a file, call an API. When the LLM decides it needs information or needs to act, it emits a tool call instruction which the MCP client executes.

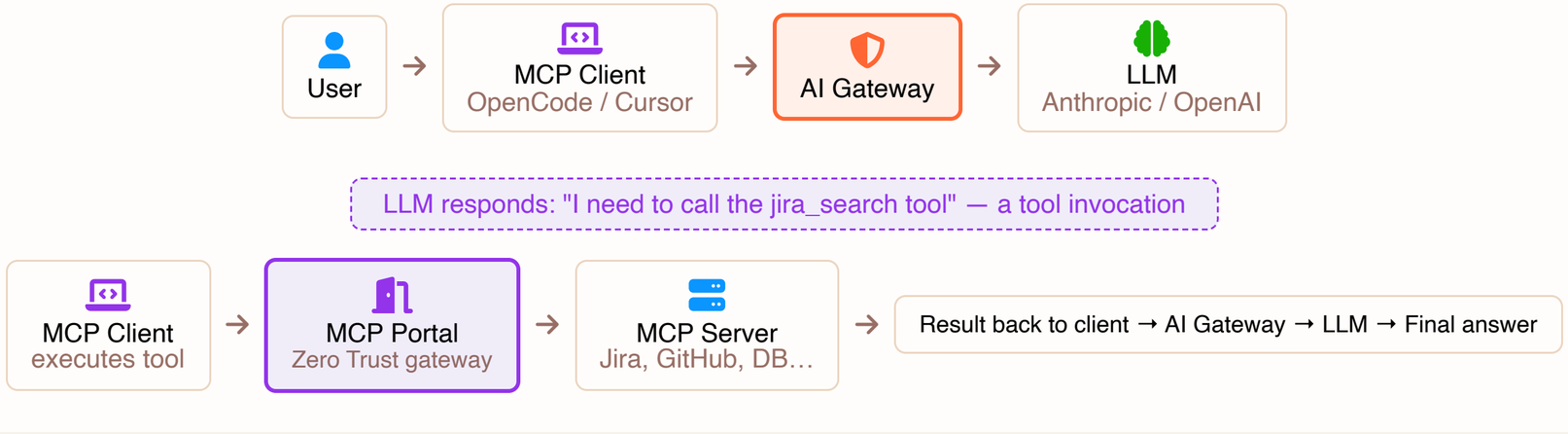
⚠ Why MCP creates new risk

- **Prompt injection via tools:** Malicious data returned from a tool can hijack the agent
- **Supply chain:** Unvetted third-party MCP servers with unknown code
- **Privilege escalation:** Agents acting with excessive permissions
- **Data exfiltration:** Agent fetches sensitive data and passes it to a compromised LLM
- **Shadow MCP:** Employees connecting to unauthorized servers

Real Example (2025): A team collaboration tool's MCP integration exposed customer data to other orgs due to a bug, forcing the integration offline for two weeks (CVE-2025-6514).

The MCP Request Loop

Understanding where each Cloudflare product sits in the agentic flow is key to the security story.



AI Gateway — sits on the LLM call path. Handles all calls to the LLM provider. Rate limiting, cost control, DLP on prompts/responses, observability.

MCP Portal — sits on the tool call path. Controls which MCP servers and tools the agent can access. Auth, logging, DLP on tool data.

MCP Server Portal: Zero Trust for Agents

Think of it as an **SSO app launcher for MCP servers**. Instead of configuring dozens of individual server URLs in an MCP client, users connect to one portal endpoint — and get access to every tool they're authorised to use.

Single Front Door

Without a portal, employees need:

- Separate URL for each MCP server
- Individual authentication per server
- Manual updates when servers change

With a portal:

- One URL configured in MCP client
- Authenticate once via corporate IdP
- New servers appear automatically

Access Control

- SSO + MFA via Cloudflare Access
- Device posture checks
- Group-based server visibility
- Least privilege: hide unused tools

Audit Logging

- Every tool call logged centrally
- Who called what, when
- Anomaly detection on usage patterns

DLP on Tools

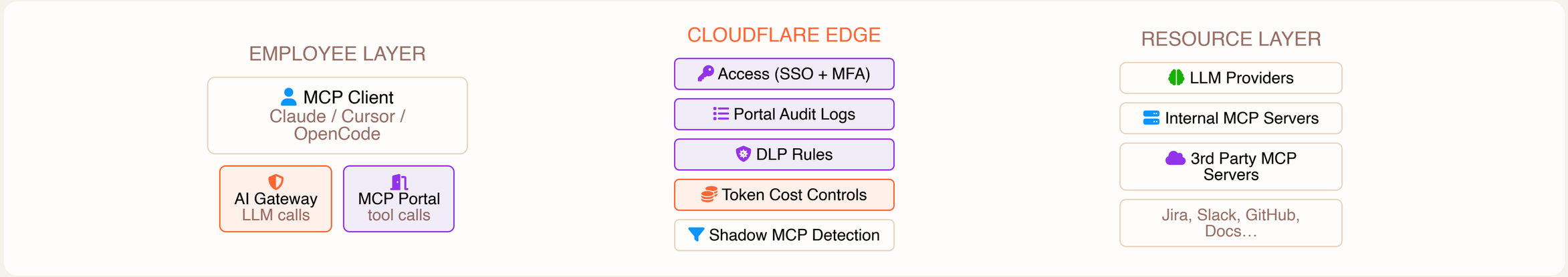
- Scan tool inputs/outputs for PII
- Prevent sensitive data passing to LLM
- Block specific tool

Code Mode

- Collapses 52 tools → 2 portal tools
- 94% reduction in token usage
- Fixed cost regardless

Enterprise MCP Reference Architecture

Cloudflare's own internal deployment — across engineering, sales, finance, and marketing teams using agentic workflows.



AI Gateway controls LLM vendor spend, prevents lock-in, provides full prompt/response logging

MCP Portal controls which employees can access which tools, logs every tool invocation

Gateway (SWG) detects Shadow MCP — employees using unauthorised servers outside the portal

Shadow MCP Detection

Employees can bypass the MCP portal and connect their AI client directly to any public MCP server. Cloudflare Gateway can detect and block this at the network level.

Detection Methods (via Gateway)

- **Known hostnames:** mcp.stripe.com, known MCP server domains
- **Wildcard patterns:** `mcp.*` subdomains
- **URL paths:** Requests containing `/mcp` or `/sse`
- **Body inspection (DLP):** JSON-RPC method signatures in HTTP body

Why body inspection matters

MCP uses JSON-RPC over HTTP. Every request contains a "method" field — even if the URL doesn't reveal it's MCP traffic. DLP regex rules can catch this.

Gateway DLP Regex Patterns

```
"method":"initialize"
"method":"tools/call"
"method":"tools/list"
"method":"resources/read"
"method":"prompts/list"
"protocolVersion":"202[4-9]
```

Gateway Actions Available

- **Log:** Visibility without blocking — discover shadow MCP first
- **Block:** Prevent access to non-portal MCP traffic
- **Redirect:** Send users to the approved portal

AI Security for Apps (WAF)

Important: This protects your own AI-powered applications or MCP servers exposed to the internet — not employees using external AI services. It inspects **inbound** traffic from external users and agents calling your endpoints.

Prompt Injection Scoring

Every inbound prompt receives a score from **1-99**

1-10	High Risk
11-50	Medium Risk
51-99	Low Risk

WAF field: `cf.llm.prompt.injection_score`

PII Detection

Two detection methods:

Fuzzy (AI): Context-aware, finds obfuscated PII

Exact (Regex): Pattern matching for SSN, CC, etc.

WAF field: `cf.llm.prompt.pii_detected`

Unsafe Topic Detection

Block harmful content categories:

- Violence & weapons
- Illegal activities
- Hate speech
- Custom topics (e.g., competitors)

WAF field: `cf.llm.prompt.flagged_topics`

Requirements: Enterprise plan · Domain proxied through Cloudflare · Endpoints labelled `cf-llm` · WAF enabled on zone

Use cases: Public-facing chatbots · Customer-accessible MCP servers · Any HTTP endpoint that accepts LLM prompts

WAF Rules for AI Security

Example: Block High-Risk Prompts

WAF Custom Rule

```
(cf.llm.prompt.injection_score lt 20)
  and
(http.request.uri.path contains "/api/chat")
```

Action: Block | Challenge | Log

Example: Alert on PII

WAF Custom Rule

```
(cf.llm.prompt.pii_detected eq true)
  and
(cf.llm.prompt.pii_types contains "ssn")
```

Available WAF Fields

`cf.llm.prompt.injection_score`	1-99
`cf.llm.prompt.pii_detected`	Boolean
`cf.llm.prompt.pii_types`	Array
`cf.llm.prompt.flagged_topics`	Array
`cf.llm.prompt.is_safe`	Boolean

Requirements

- Enterprise plan required
- Label endpoints with `cf-llm` content type
- Works with any LLM backend

Pillar 3: Customer Benefits

Governed MCP Access

- Single portal — one URL for all approved MCP servers
- Only vetted servers visible to employees
- Identity-enforced access per user/group
- No Shadow MCP with portal enforcement

Token Cost Reduction

- Code Mode collapses tool schemas into 2 portal tools
- 94% reduction in context token usage
- Cost stays fixed as more MCP servers are added
- AI Gateway enforces per-user token budgets

Inbound App Protection

- WAF protects your public AI endpoints
- Prompt injection scoring on every request
- PII detection and topic filtering
- No WARP or SWG required — WAF only

Full Agentic Visibility

- Every tool call logged via MCP Portal
- Every LLM call logged via AI Gateway
- Shadow MCP detected via Gateway DLP
- Complete audit trail of agent actions

Summary: Three Pillars of AI Security

Pillar 1
End-User Protection

Who: Employees using ChatGPT, Claude, Copilot in browsers

Solution:

- Cloudflare One + WARP
- Gateway (SWG) + DLP
- Browser Isolation
- CASB for shadow AI discovery

Outcomes:
Shadow AI visibility, PII blocking, compliance

Pillar 2
App & API Security

Who: Developers, applications calling LLM APIs

Solution:

- AI Gateway
- DLP on prompts & responses
- Dynamic routing & fallbacks
- Rate limiting & cost controls

Outcomes:
Cost control, resilience, observability

Pillar 3
Agentic & MCP Security

Who: AI agents, MCP clients, public AI apps

Solution:

- MCP Server Portal (Access)
- AI Gateway for LLM calls
- AI Security for Apps (WAF)
- Gateway for Shadow MCP detection

Outcomes:
Zero Trust tools access, audit trail, inbound protection

All three pillars managed from a single Cloudflare dashboard — Zero Trust, AI Gateway, and WAF

Getting Started

Pillar 1

1. **Deploy WARP**
   Route employee traffic through Cloudflare
2. **Enable AI App Controls**
   Select AI apps from library, set policies
3. **Configure DLP Profiles**
   Define PII patterns, set block/warn actions
4. **Monitor & Refine**
   Review logs, tune policies

Pillar 2

1. **Create AI Gateway**
   Set up gateway in dashboard
2. **Point Apps to Gateway**
   Replace provider URLs with gateway endpoint
3. **Enable Guardrails**
   DLP, rate limits, budget caps
4. **Build Routing Flows**
   Failovers, A/B tests, conditionals

Pillar 3

1. **Create MCP Portal**
   Access > AI Controls in Zero Trust dashboard
2. **Register MCP Servers**
   Add internal & 3rd-party MCP servers
3. **Set Access Policies**
   Who can access which servers/tools
4. **Enable WAF for Public APIs**
   Label `cf-llm` endpoints, configure rules

Questions? Let's discuss your AI security requirements.