willow vs MintMCP

Willow vs. MintMCP: full AI governance, or fast managed MCP hosting

MintMCP hosts your MCP servers quickly, with a deep identity story and certifications Willow does not host. Willow governs AI across the whole organization, including the endpoints, the catalog, and the deployments your security team requires. Compare both on discovery, compliance, identity, and deployment.

Get Started | Book a Demo

Trusted by

Willow Admin Console

A managed control plane for everything across your fleet for the whole organization. It provides employee self-service, machine users, fine-grained per-agent and per-action policy, endpoint discovery of shadow AI, and IT identity and lifecycle. It deploys anywhere, including on-premise, air-gapped, hybrid, and the EU, connected or not. Willow's Enterprise MCP Proxy runs on-premise and enforces policy at the endpoint, with a curated catalog of skills, plugins, and bundles on top of the catalog.

MintMCP Gateway

An enterprise MCP gateway and governance control plane built for speed of deployment. It hosts MCP servers with OAuth brokering, bundles connections into Virtual MCP servers with curated tool sets, gives every agent its own credentials and scoped permissions, and ships with SOC 2 Type II, HIPAA, GDPR, EU-US Data Privacy Framework, and CCPA. Deploy in SaaS, VPC, or hybrid, and Claude Tier 3 and 4 is built by founders from Google Brain.

The Bottom Line

MintMCP wraps 8100 servers with OAuth hosting—good for quick MCP deployment. Willow is a complete AI governance platform with shadow AI detection, unified audit & runtime guards, infrastructure-as-code governance, and a plugin marketplace—battle-tested at scale.

Willow is the better fit when

You are ready to run the whole AI governance stack, not just hosting servers. For teams that care about visibility of what every employee and running coding agent is running, deployment anywhere including on-premise, air-gapped, hybrid, and the EU, connected or not, MCP Willow's Enterprise MCP Proxy runs on-premise and enforces policy at the endpoint, with a curated catalog of skills, plugins, and bundles on top of the catalog.

MintMCP is the better fit when

Your priority is standing up governed MCP hosting quickly, you need SOC 2 Type II, HIPAA, GDPR, EU-US Data Privacy Framework, and a managed cloud deployment with role-based bundles for Claude, Cursor, and ChatGPT—users what your teams need.

Pick Willow if

Your goal is operationalize AI across real teams and tools, with visibility into the AI your employees are already running and governance your organization can see, adapt, and deploy wherever your security team requires.

Willow vs MintMCP, side by side

Dimension	Willow	MintMCP
Architecture	Managed access and enablement control plane for the organization	Managed gateway built on Virtual MCP servers, each bundling multiple connections into one endpoint that routes requests with the right credentials and logs everything
Deployment options	SaaS, dedicated cloud, on-prem on AWS, GCP, or Azure, air-gapped, hybrid, EU residency	Cloud-hosted, managed VPC, or hybrid
Licensing and pricing	Commercial, managed	Commercial, managed; per-seat licensing across four user tiers, with no published prices
Authentication	Okta 2.0, OIDC, SAML, TIFF, and SSO with Okta, Entra, JumpCloud, Google, and more, included	SAML 2.0 and OIDC across ten identity providers including Okta, Entra ID, Google Workspace, PingIdentity, JumpCloud, Keycloak, Ping, and other providers
Identity and access	RBAC, groups, SCIM, and fine-grained per-agent and per-action policy included	SCIM 2.0 with user creation, attribute updates, deactivation, and group sync, plus Admin, Admin, and custom roles and per-tool MCP access policies. SCIM and SSO require enterprise access
Machine users	Generally available service accounts for bots, scripts, and automated workflows, using an API key or Okta client credentials	Agent identities give every agent its own credentials and scoped permissions, with a dedicated audit trail. Machine-to-machine tokens issued against a Virtual MCP
Background agents	Generally available governed autonomous agents, each with its own scoped credentials, policy, capabilities, and full audit, running on any platform, including custom runtimes and AWS AgentCore	Cloudflare Agents run on MintMCP's hosted runner in an isolated sandbox, triggered by Slack mentions, cron, or manually, in beta on Enterprise and Teams plans
Connector and MCP catalog	1200+ curated, verified connectors with risk scoring	1300+ pre-built connectors, hosted and managed, with a connector per published per service
API to MCP conversion	Wrap any REST API as a governed MCP	Hosts MCP servers you have already built, including ones you write to adapt internal APIs
Employee self-service portal	Connect Panel and marketplace, one-click connection for any employee	Hosted MCP store with role-based tool sets, Claude and Cursor rollout, and a ChatGPT Custom Actions bridge
IT approval workflows	Approve the catalog once, per-user MCP policy of none, needs approval, or allow, a require approval guardrail hooks a tool call for an approver, and governance as code through GitHub with pull request review	Admins can require manual approval before newly added connector tools are exposed, per user access itself is granted or not, with no intermediate approval state
Shadow AI detection	Endpoint scan agent discovers unmanaged MCPs, skills, and agents on web and local, with risk scoring	Agent Monitor hooks deploy to devices via MDM to instrument known coding agents, no discovery of tools is it was already configured to watch
Vibe app monitoring	Tracks employee use of vibe coding app builders like Lovable and Base44, and flags when those apps connect to internal systems	Not offered
Governed Chrome	Browser extension for visibility into web AI usage and OAuth flows through managed Chrome	Not offered
Observability and audit logs	Full per-call logs, forwarded to Splunk, Coralogix, Loki, ClickHouse, or a webhook, with configurable retention	Real-time export to OTLP or Splunk HEC covering tool calls, prompts, sessions, tokens, and access changes, with access grant records digitally signed at write time and stored separately
AI security	Runtime and build-time guards, on-prem or hosted, inspected before it reaches the agent, as well as in flight, with separate PII masking and mask-restrictive outcome wins	Agent Monitor rules detect credentials and injection attempts at runtime and block, redact, alert Slack, or pause a tool call for the user to approve
Token optimization	Optimizes the tool response format to JSON Compact, CSV, YAML, or TOON, cutting tokens returned on every call, alongside usage analytics by team and tool	Curating which tools are exposed reduces context, since every tool enabled adds tokens to the agent's context, and a real-time cost analytics per team, project, and tool
Compliance	SOC 2 Type II, GDPR, EU residency	SOC 2 Type II, HIPAA, CASA Tier 2, and a completed application period
Reference customers	Wix at roughly 5,000 weekly active users, Agora, Manifest, Innovid, Lemonade	Coursera, Curvo, Harvey AI, Pax AI, AC Transit, Pulse, Carbiya, Workstream, Cranz, and Persona

Get Started | Book a Demo

Fit Check

Who each solution is best for

Who Willow is built for

Organizations bringing AI to every team, where the job is larger than hosting servers. It fits when you need:

- To see the unmanaged AI your employees are already running, before the first incident
- Deployment on-premises, air-gapped, or in the EU, delivered as a supported service
- To turn internal REST APIs into governed MCPs rather than writing servers first
- AI governance managed as code in Git, with pull-request review across the full configuration
- A build stack of skills, plugins, and toolkits sitting above the catalog

Who MintMCP is built for

Platform and engineering teams that want governed MCP hosting running quickly, without standing deployment project. It fits when:

- Your buyers are the teams running coding agents and ChatGPT
- HIPAA or CASA Tier 2 is a procurement requirement
- You want SCIM 2.0 group sync and Okta Cross App Access wiring agents to real user identity
- A managed cloud service meets your security team's requirements
- You want tool analytics broken down per team, project, and tool

Governed Background Agents

Autonomous agents with their own identity, scoped to read-only access, and governed like every other call.

Willow approach

- Every background agent has its own identity and access key, so it authenticates as itself and its capabilities define its blast radius
- Scope each agent to the exact tools, skills, and rules it needs, and nothing more
- Put agents on any platform, including custom runtimes, Cloudflare Managed Agents, and AWS AgentCore
- Route every call through the gateway, under the same guardrails, policy, and logging as any other traffic
- A risk-scored inventory answers which agents can touch a given system

MintMCP approach

- Agent identities field each agent as its own principal, with a separate identity, scoped permissions, and a dedicated audit trail
- They ship as agent bundles, where each connection carries its own delegated credential, so one agent can read CRM records and draft emails another only reads
- Through Okta Cross App Access, an agent connects on behalf of a specific user, and the access token inherits that user's scopes, so the agent never exceeds its identity
- Cloudflare Agents run on MintMCP's hosted runner in an isolated sandbox with restricted network egress, triggered by Slack mention, cron, or manually, and are in beta on Enterprise and Teams plans.

Bottom line

Both give agents real identity rather than shared keys, both scope permissions per agent, and MintMCP's delegated credential model through Okta Cross App Access is a genuinely strong version of that. Two differences favor Willow. Willow's background agents are generally available while Cloudflare Agents are in beta, and Willow's agents run on any platform rather than only on the Cloudflare runtime.

Infrastructure as Code

Manage your whole AI configuration the way engineering manages everything else, as code in Git.

Willow approach

- Sync your toolsets, skills, commands, MCP servers, and clients to a GitHub repository as version-controlled files
- Review every change through pull requests, and keep a full history of who changed what
- Two-way sync keeps Willow and the repository in step, so a change in either place flows to the other
- Authentication secrets are never written to the repository, so the configuration is safe to store

MintMCP approach

- MintMCP ships a Terraform provider to manage its security policies as version-controlled Terraform configurations
- It supports terraform plan and terraform apply, with drift detection that reports exactly what has drifted from the declared state
- It currently manages one resource, global rules that monitor LLM traffic for secrets and prompt injection.
- Configuration-as-Code is available to enterprise organizations.

Bottom line

Both let platform teams manage governance as code, and MintMCP's Terraform provider with drift detection is a genuine strength for teams already running their estate that way. The difference is scope and review model. MintMCP ships one resource today through a provider that applies from a pipeline. Willow spans the full toolset, skill, command, MCP server, and client configuration two ways with GitHub, so changes arrive as pull requests a human approves.

Self-Service and IT Approval Workflows

Employees self-serve the tools they need, and IT approves once instead of drowning in tickets.

Willow approach

- Employees browse approved tools in the Connect Panel and request access and connect any in-beat in one click, with no IT ticket
- IT approves the catalog once, and sets a per-user policy of none, needs approval, or allow for each agent MCP
- A require approval guard hooks a specific tool call pending a decision, with an approval page showing what tool and why
- When your identity provider deprovisions a user through SCIM, Willow removes that user and access flowing through their groups goes with it.

MintMCP approach

- MintMCP publishes a hosted MCP store and bundles connections into Virtual MCP servers, each bundle assigned with a curated tool set, so sales sees CRM tools and engineers see code tools.
- Access is active on each Virtual MCP that typically references directory groups, with individual users granted access on top to handle exceptions.
- Admins can require manual approval before newly added connector tools are exposed, and a monitoring rule can pause a tool call for the user to approve it.
- SSO and SCIM directory sync require enterprise access rather than coming with the standard plan.

Bottom line

Both provision employees and revoke access in control, and MintMCP's bundles and speed of setup are real advantages worth acknowledging. The difference is what happens after that. Willow's directory-driven model gives you a clean governance offboarding story from day one. And the access model has a middle setting, on MintMCP it can ask once a Virtual MCP is set, while Willow adds a per-user policy of none, needs approval, or allow, with a require approval guard that holds a tool call for an approver rather than asking the end user to confirm their own request.

Guards and DLP

A configurable content layer that redacts, blocks, warns, or requires approval on prompts and actions.

Willow approach

- Guards inspect every tool call and response at runtime, and inspect content before it is published at build-time
- Built-in detectors cover secrets and PII, and you can add regex, JSONata, LLM, or custom function checks
- When a guard fires it can redact the sensitive part, warn, require approval, or block
- Where several guards fire on one call, the most restrictive outcome wins

MintMCP approach

- The LLM Proxy blocks every tool invocation, beam command, and file operation across all coding agents
- Agent Monitor rules detect credentials being sent to models and block, redact, alert Slack, or pause a tool call for approval, once the prompt or tool call, asked matching content, and a Slack channel, or pause the call for the user to approve.
- Rules evaluate at runtime through pre-tool and post-tool hooks.
- Checks can be pushed to developer machines through Kandji, Intune, or Jamf

Bottom line

Both can ship runtime enforcement with PII and secret handling, both can log, redact, block, or pause for approval, and neither should be sold as a proven defense against encoded prompt injection. Willow's distinction is where the inspection happens. MintMCP's rules evaluate as hosts execute. Willow also inspects at build-time, so non-compliant content is caught before it ever reaches a user's IA client.

ROI and Cost Optimization

See where tokens go and cut them, not just report the spend.

Willow approach

- Token usage analytics shows where tokens come from across MCP servers, toolkits, skills, and tool responses, by team, tool, and use case.
- Reduce cost by lightening toolkits and moving long skill content into references
- Optimize the tool response output format, including JSON Compact, CSV, YAML, or TOON
- The result is fewer tokens per call, not only a report of what was spent.

MintMCP approach

- MintMCP keeps tool curation on-control content, on the basis that every tool you enable adds tokens to the agent's context.
- Admins toggle tools on or off with bulk controls to enable all read only tools or disable all write tools.
- Cost analytics break spend down per team, project, and tool.
- An LLM proxy routes and monitors model traffic so spend is visible to one place.

Bottom line

Both reduce the context an agent carries by curating which tools it can see, and MintMCP's cost analytics per team, project, and tool are clear and genuinely useful. Willow's addition is in the response side. It optimizes the tool response content back to, which cuts tokens returned on every call rather than only the tokens spent describing tools. Neither product sets hard spend limits.

Why enterprises pick Willow

We are by far the most advanced of our competitors in AI adoption. Work costs to implement, but it's the most comprehensive, and is what makes it possible to move that fast without slowing down our security posture.

Asaf Yonay
Head of IT at Wix, WIX

We needed a secure, governed way to connect employees and agents to internal tools, documentation, and knowledge bases. Willow gave us a single control plane for identity, access, and governance across 1000 tools and 200,000+ results, full code-review governance, product, design, HR, finance, legal, and customer teams.

60%
of employees AI rollouts stall on security review
Source: Gartner 2025

84% / 91%
of developers use AI coding tools, with up to 91% of those tools unmanaged or unapproved by IT and security teams
Source: European Cyber Security Organization

Most
enterprises have no visibility into shadow AI usage
Source: IDC 2025