AVIATRIX

TECHNICAL BRIEF

Contain AI Agent Harnesses — OpenClaw / NemoClaw

Powered by Aviatrix Cloud Native Security Fabric

Terraform reference architecture for private agent hosts, Aviatrix egress control, monitor-first rollout, and audit evidence

Threat Context

OpenClaw, Hermes, and NemoClaw agent harnesses are persistent, privileged non-human operators. Once deployed on a VM, Kubernetes cluster, or cloud sandbox, they become outbound network actors with delegated authority. By default they have unrestricted outbound internet access. Their tool calls, skill invocations, browser sessions, and package installs are indistinguishable on the wire from legitimate work unless a control at the network layer explicitly limits what they can reach. Four threat vectors are in scope for this Validated Containment Architecture:

- Prompt injection, malicious skill, or compromised runtime redirecting the agent to exfiltrate data to an attacker-controlled endpoint (OWASP LLM01, LLM05).
- Agent calling an unsanctioned model provider or unapproved API carrying sensitive data retrieved from sanctioned sources (OWASP LLM08 excessive agency).
- Direct DNS exfiltration: encoded data is sent through an unapproved external DNS resolver over UDP or TCP port 53.
- Compromised agent spoke reaching adjacent workloads and expanding Blast Radius beyond the agent host.

REFERENCE VALIDATION SCENARIOS

The scenario set validates the architecture against the OWASP LLM Top Ten and MITRE ATLAS in a live deployment. LLM01 prompt injection driving exfiltration is closed by default-deny on the attacker domain: the socket does not open. LLM05 malicious skill or supply-chain dependency phoning home is closed by the same default-deny policy. LLM08 shadow model call is closed by the named vca-openclaw-shadow-model-deny rule, evaluated ahead of all allow rules and logged by name in CoPilot FlowIQ. DNS-tunneled exfiltration is closed by the allow-vpc-dns rules followed by the deny-dns-exfil rules, which permit the VPC resolver and deny all external UDP/TCP 53. Optional DCF east-west policy can deny routed access to configured internal CIDRs. This is a phase-two control and is not required for the initial egress-only deployment. Every blocked flow is visible in CoPilot FlowIQ with a human-readable rule name.

This Technical Brief describes the AWS implementation pattern for the OpenClaw Agent Runtime Containment VCA. The blueprint creates a private agent subnet, routes outbound traffic through an Aviatrix Spoke Gateway, applies ordered Distributed Cloud Firewall (DCF) policy, and logs every decision in CoPilot FlowIQ. Terraform provisions the workload network, the Spoke Gateway, SmartGroups, WebGroups, ordered DCF policies, CoPilot/syslog hooks, examples, and test scenarios. The VM shape is the primary scope; the same SmartGroup/WebGroup model extends to Kubernetes namespaces.

REFERENCE STATUS

This is a reference blueprint. It has been Lab-validated against a live Aviatrix Controller and AWS account. Run terraform `fmt/validate/plan` and a monitor-mode deployment in your own environment before production use. Run terraform fmt, validate, plan, and a non-production monitor-mode deployment before production use.

Insertion Pattern

The private agent host has no public IP and no direct internet route. SSM Session Manager replaces public SSH. The subnet default route sends 0.0.0.0/0 to the Aviatrix Spoke Gateway, which becomes the transparent egress enforcement point.

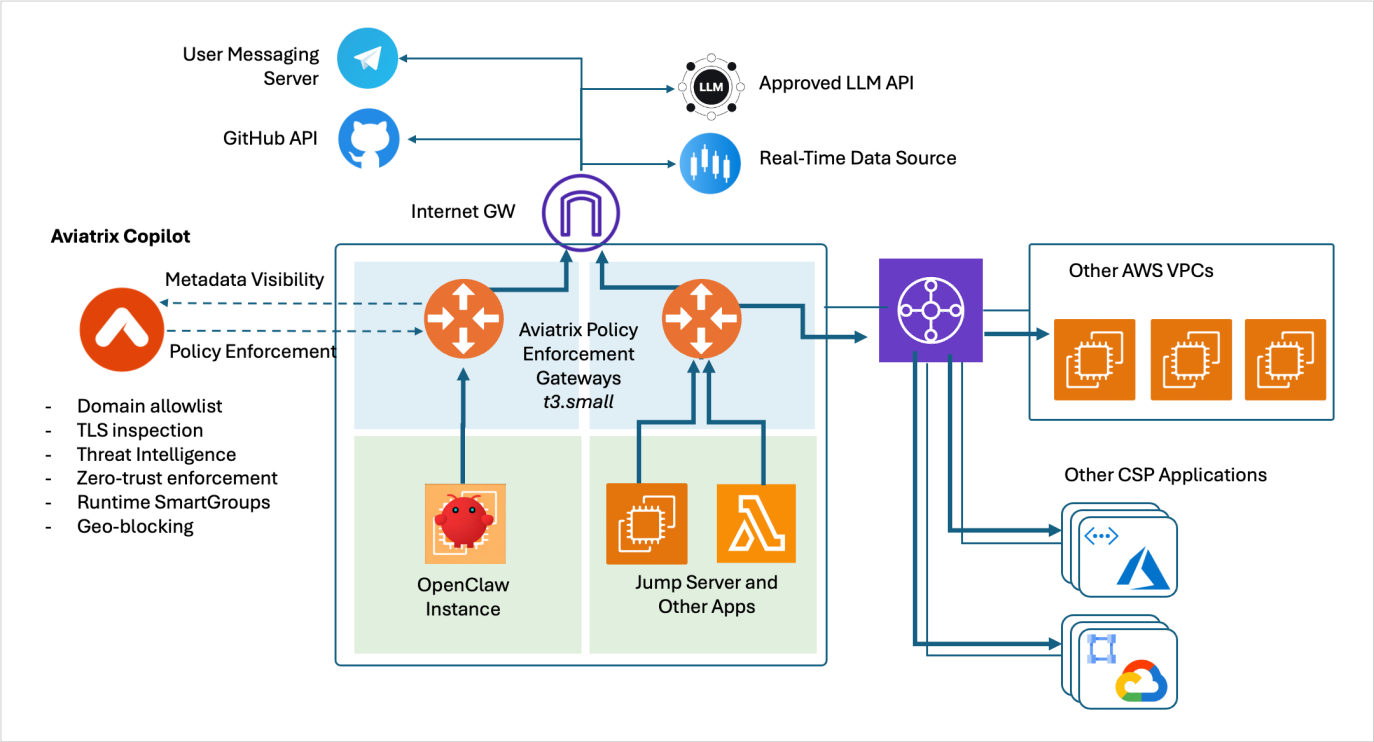


Figure 1. Reference insertion point for AWS VPC egress and east-west control.

Resource	Default implementation
VPC and subnets	Dedicated VPC with a private agent subnet, a public gateway subnet for Aviatrix, route tables, and VPC Flow Logs to CloudWatch.
Agent host	EC2 VM for OpenClaw/Hermes/NemoClaw – no public IP, least-privilege instance profile, encrypted root volume, IMDSv2 required.
Management path	Interface endpoints for ssm, ssmmessages, ec2messages, plus CloudWatch Logs and STS endpoints for private operation; optional S3 gateway endpoint.
Aviatrix Spoke Gateway	Attached to the VPC via terraform-aviatrix-modules/mc-spoke. The agent subnet default route points to the gateway.
DCF objects	SmartGroups for agent subnet and destinations; WebGroups for approved FQDN sets; an ordered DCF policy list and a POST_RULES default action.

Prerequisites

The private agent host has no public IP and no direct internet route. SSM Session Manager replaces public SSH. The subnet default route sends 0.0.0.0/0 to the Aviatrix Spoke Gateway, which becomes the transparent egress enforcement point.

- Aviatrix Controller 8.1+ for FQDN domain policy and SmartGroups; Controller/provider 8.2+ for the POST_RULES default-action resource; 9.0+ if selective TLS decryption or egress IDS/IPS is added later.
- CoPilot connected to the Controller with DCF visibility and FlowIQ ingestion enabled.
- AWS account onboarded to Aviatrix, with an agent VPC where the Spoke Gateway can be deployed.
- Terraform 1.5+ and AWS credentials for VPC, EC2, IAM, CloudWatch Logs, VPC endpoints, and Flow Logs.
- A policy-as-code repository where WebGroup changes can be reviewed by pull request.

SmartGroup and WebGroup Design

SmartGroups classify workloads by identity like CSP tags, type, account, region etc. Membership updates automatically at runtime: tag a new agent and it inherits the policy the moment it deploys, with no rules to rewrite. That dynamic, identity-following enforcement is what sets Aviatrix apart from static IP- and subnet-based firewalls.

Object	Type	Purpose
sg-agent-workload	SmartGroup - CSP tag (type = vm)	Source identity for the agent workload. Matches the VM by CSP tag (default Role = openclaw-agent-harness), so policy follows the workload regardless of IP or subnet. Tag key/value configurable via agent_workload_tag_key / agent_workload_tag_value.
sg-vpc-dns-resolver	SmartGroup - CIDR	VPC DNS resolver (.2/32). Permitted before the external-DNS deny so normal name resolution keeps working.
sg-east-west-deny	SmartGroup - CIDR	Adjacent / internal RFC1918 CIDRs. Destination for the east-west (lateral-movement) deny rule.
sg-approved-model-gateway-cidrs	SmartGroup - CIDR *(optional)*	Internal model-gateway CIDR destinations. Created only when approved_model_gateway_cidrs is set.

WebGroups are named sets of approved destinations matched by TLS server name (SNI), exact hosts or wildcard domains like *.github.com, and, with TLS decryption, by URL path. It is grouped by intent so one policy stays readable, and each agent class enables only what it needs.

Object	Type	Purpose
wg-aws-infra	WebGroup - Domains	Region-specific AWS control-plane endpoints (SSM, EC2, STS, CloudWatch Logs, ECR, S3), generated from `aws_region`. Always created.
wg-openclaw-core	WebGroup - Domains	OpenClaw / NemoClaw / Hermes core services, docs, and required terminal endpoints.
wg-approved-model-gateways	WebGroup - Domains	Sanctioned model / inference endpoints (NVIDIA by default). Prefer an enterprise model gateway. Created when `approved_model_gateway_domains` is set.
wg-package-registries	WebGroup - Domains	npm, PyPI, GitHub, Hugging Face, Docker - coding-class agents. Created when `enable_package_installs` is true.
wg-os-updates	WebGroup - Domains	HTTPS OS / update mirrors for staged installs. Created when `os_update_domains` is set.
wg-approved-saas-apis	WebGroup - Domains	Business SaaS APIs approved per agent class (CRM, ticketing, document stores, workflow). Created when `approved_saas_api_domains` is set.
wg-approved-mcp-gateways	WebGroup - Domains	Enterprise MCP / tool gateways without flat internal reachability. Created when `approved_mcp_gateway_domains` is set.
wg-identity-telemetry	WebGroup - Domains	Approved identity-provider and observability / telemetry endpoints. Created when `identity_and_telemetry_domains` is set.
wg-public-reference	WebGroup - Domains	Search / weather / reference endpoints for demo or research only. Created when `enable_public_reference` is true.
wg-unapproved-model-providers	WebGroup - Domains	Known unapproved model APIs (OpenAI, Anthropic, Mistral, DeepSeek, …). Denied ahead of all allow rules so shadow-model attempts are visible.

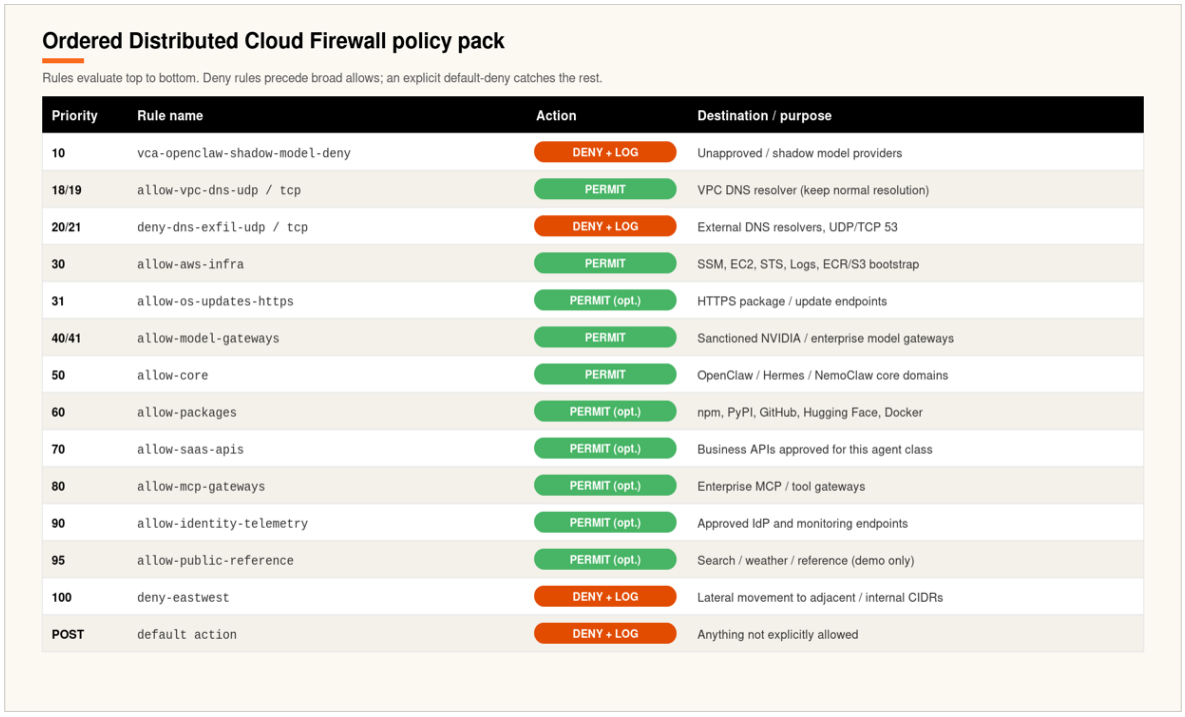
KUBERNETES EXTENSION

For Kubernetes, map agent pods to namespace or label-based SmartGroups and apply the same destination WebGroups. Policy follows the pod identity instead of relying on static pod IPs.

Rollout principle: Deploy in monitor mode first, exercise real terminal workflows, convert observed legitimate destinations into WebGroups by pull request, then switch to enforce mode.

Distributed Cloud Firewall Policy Pack

Rules evaluate top to bottom. A plan-time guardrail blocks the same provider appearing in both the approved and unapproved model lists. Use domain-based WebGroups for HTTP/HTTPS/TLS destinations; supported FQDN wildcards may be used. Keep ports in the DCF rule.



Ordered Distributed Cloud Firewall policy pack

Rules evaluate top to bottom. Deny rules precede broad allows; an explicit default-deny catches the rest.

Priority	Rule name	Action	Destination / purpose
10	vca-openclaw-shadow-model-deny	DENY + LOG	Unapproved / shadow model providers
18/19	allow-vpc-dns-udp / tcp	PERMIT	VPC DNS resolver (keep normal resolution)
20/21	deny-dns-exfil-udp / tcp	DENY + LOG	External DNS resolvers, UDP/TCP 53
30	allow-aws-infra	PERMIT	SSM, EC2, STS, Logs, ECR/S3 bootstrap
31	allow-os-updates-https	PERMIT (opt.)	HTTPS package / update endpoints
40/41	allow-model-gateways	PERMIT	Sanctioned NVIDIA / enterprise model gateways
50	allow-core	PERMIT	OpenClaw / Hermes / NemoClaw core domains
60	allow-packages	PERMIT (opt.)	npm, PyPI, GitHub, Hugging Face, Docker
70	allow-saas-apis	PERMIT (opt.)	Business APIs approved for this agent class
80	allow-mcp-gateways	PERMIT (opt.)	Enterprise MCP / tool gateways
90	allow-identity-telemetry	PERMIT (opt.)	Approved IdP and monitoring endpoints
95	allow-public-reference	PERMIT (opt.)	Search / weather / reference (demo only)
100	deny-eastwest	DENY + LOG	Lateral movement to adjacent / internal CIDRs
POST	default action	DENY + LOG	Anything not explicitly allowed

Figure 2. Ordered DCF policy pack for the agent runtime.

Priority	Rule name	Action	Destination / protocol
10	vca-openclaw-shadow-model-deny	Deny + log	wg-unapproved-model-providers
18/19	allow-vpc-dns-udp / tcp	Permit	sg-vpc-dns-resolver UDP/TCP 53
20/21	deny-dns-exfil-udp / tcp	Deny + log	Any UDP/TCP 53 except VPC resolver
30	allow-aws-infra	Permit	SSM, EC2, Logs, STS, ECR/S3 endpoints
40/41	allow-model-gateways	Permit	sg-approved-model-gateways TCP 443
50	allow-core	Permit	wg-openclaw-core TCP 443
60	allow-packages	Permit (opt.)	wg-approved-package-registries TCP 443
70/80	allow-saas / mcp	Permit (opt.)	Approved SaaS APIs and MCP gateways TCP 443
100	deny-eastwest	Deny + log	Adjacent / internal CIDRs
POST	default action	Deny + log	Any unmatched destination

Agent-Class Presets

Agent class	Typical permits	Typical denies
Locked-down demo	Model gateway, OpenClaw/ NemoClaw core, AWS private ops.	Package registries, broad web, SaaS APIs.
Coding agent	GitHub, package registries, internal artifacts, model gateway, docs.	Shadow models, external DNS, unapproved SaaS, production networks.
Research agent	Approved search/data APIs, document stores, model gateway, telemetry.	Package registries unless needed, arbitrary uploads.
Support agent	CRM, ticketing, approved KB, MCP gateway, model gateway.	Source-code systems and package registries unless required.
Regulated-data agent	Specific internal APIs, approved model gateway, observability.	Public package registries, public SaaS, external upload, shadow models.

Validation Tests

Test	Expected result
Allowed model call	Agent reaches the approved model gateway; FlowIQ logs the permit rule.
OpenClaw terminal launch	Core endpoints resolve and connect; no public SSH required; SSM session works.
Blocked attacker domain	Connection fails; FlowIQ shows default-deny or a named deny.
Blocked shadow model	Connection to an unapproved provider fails; FlowIQ shows the shadow-model deny.
Blocked external DNS	dig/curl via 8.8.8.8 or 1.1.1.1 on UDP/TCP 53 fails; the VPC resolver still works.
Monitor-to-enforce update	A new legitimate destination appears as would-be-denied, is added by PR, then succeeds after apply.

Promote with the standard sequence:

```
cp terraform.tfvars.example terraform.tfvars   # policy_mode = "monitor"
make preflight && terraform plan && terraform apply
POLICY_MODE=enforce /opt/openclaw-vca/verify-egress.sh
```

Known Constraints

Constraint	Recommended handling
Strict allow-lists can break first boot.	Disable automated package bootstrap in enforce mode; use monitor mode or a pre-baked AMI.
Some package mirrors use HTTP or non-SNI flows.	Prefer internal artifact stores and HTTPS-only endpoints; many Ubuntu apt mirrors are HTTP.
Endpoint IPs and SaaS domains change.	Keep WebGroups in Terraform and review updates by pull request.
TLS decryption is sensitive.	Use only when URL-path enforcement is required; scope narrowly and retest certificate pinning.
Policy-list ownership can be global.	Set manage_controller_policy=false or merge DCF policy centrally to avoid clobbering other blueprints.

Appendix

Default terminal workflow destinations

The starting domain catalog is intentionally small and class-based. A locked-down demo agent may need only the model gateway and OpenClaw/NemoClaw core. A coding agent additionally needs GitHub, npm, PyPI, and internal artifact stores. A support agent needs ticketing and CRM APIs but not package registries. The default is not developer internet; it is a declared set of workflow endpoints.

WebGroup	Default FQDNs (starting catalog)
OpenClaw / NemoClaw core	openclaw.ai, www.openclaw.ai, docs.openclaw.ai, clawhub.ai, www.nvidia.com
Approved model gateways	integrate.api.nvidia.com, inference-api.nvidia.com
Package / source-control	registry.npmjs.org, pypi.org, files.pythonhosted.org, github.com, *.githubusercontent.com, huggingface.co, registry-1.docker.io
AWS infrastructure	ec2 / ssm / ssmmessages / ec2messages / logs / sts / s3 / ecr .<region>.amazonaws.com (generated from aws_region)
Shadow-model deny set	api.openai.com, api.anthropic.com, generativelanguage.googleapis.com, api.mistral.ai, openrouter.ai, api.deepseek.com, …

Aviatrix Validated Containment Architecture for OpenClaw, NemoClaw/OpenShell, and Hermes Agents removes the unknown from agentic deployment by enforcing policy at the network layer.

Ask your Aviatrix account team for a guided deployment.

Explore Validated Containment Architectures for other AI platforms.

About Aviatrix

Aviatrix® is pioneering the Cloud Native Security Fabric – the architecture the Containment Era requires. The Cloud Native Security Fabric governs every workload communication path across every cloud, every VPC, every Kubernetes cluster, and every serverless function, from a single policy plane. One rule. Universal propagation. Enforced at the workload, not at a chokepoint. Trusted by more than 500 of the world's leading enterprises. For more information, visit aviatrix.ai.