



MCP
Dev Summit
Toronto

NETCLAW

MCP at the Center of Agentic
Network Operations



John Capobianco

Head of AI & DevRel, Itential

MCP reached network automation in 131 days

John’s path from the open standard to a pyATS MCP server | November 2024–April 2025

NOV 25, 2024

ANTHROPIC RELEASES MCP

An open standard connects AI assistants to the systems where data and tools live.



MAR 17, 2025

FIRST SERVER + CLIENT

“My first working MCP Server and client... exhilarating. I see it now—why this is the hype.”



MAR 18, 2025

MCP ↔ MCP

Packet Copilot → GitHub, plus the first demo: “No REST APIs. No hardcoding.”



MAR 19, 2025

AN INTERNET OF AI AGENTS

MCP is framed as infrastructure for networks of agents—not merely tool calling.



MAR 27, 2025

“BETTER THAN I THOUGHT”

Dynamic tool discovery and capability exposure become the breakthrough.



APR 5, 2025

pyATS MCP SERVER

Network automation becomes a first-class MCP use case.

Sources: Anthropic MCP launch; John Capobianco LinkedIn posts; PulseMCP pyATS listing.

The ripple spread from projects to platforms

Infrastructure MCPs → industry adoption → NetClaw → Cisco ships official MCPs

APR–JUN 2025

JOHN BUILDS THE INFRA LAYER

- pyATS + Packet Copilot
- Cisco ACI + ISE



MAY 19, 2025

WINDOWS BECOMES AN MCP PLATFORM

Microsoft announces native MCP support across Windows.



DEC 9, 2025

MCP MOVES TO THE AAIF

The Linux Foundation launches the Agentic AI Foundation with MCP as a founding project.



MAY 20, 2026

MCP ENTERS CISCO CERT BLUEPRINTS

- CCNA v2.0 published
- Testing starts Feb 3, 2027



FEB 2026

NETCLAW FIRST COMMIT

The portfolio converges into one agentic network-ops system—the launch point for the rest of this deck.



SEP 2026

CISCO SHIPS OFFICIAL MCPs

- Meraki Dashboard
- Catalyst Center

Sources: John Capobianco projects; Microsoft Build; Linux Foundation; Cisco public repos/blueprints; NetClaw README.

The Last Mile

Why networks are the hardest place to deploy an AI agent



High-stakes

A wrong command drops traffic for everyone behind the device, not for one user.



Stateful

Sessions, routing tables and adjacencies change by the second. The agent has to read state before it acts.



Unforgiving of Hallucination

A confident but wrong BGP explanation costs more than no answer at all.



NetClaw's Answer

MCP skills that read real state, show their evidence, and wait for approval before any change.

The Journey

February 2026 to Now

675

GitHub Stars

186

Forks

233

Skills

173

MCP Integrations

**First commit in February 2026. Seven months later:
release 1.1.0 and an IETF Internet-Draft.**

From junior network engineers to multi-CCIE operators, Claw builders creating videos, content, and personalized implementations every day.

What is OpenClaw?

The Foundation of Agentic AI

Not a chatbot.

Not a copilot.

An autonomous coworker.



Released November 2025
by Peter Steinberger



60,000 GitHub stars in
days - fastest-growing
open source project in
history



Jensen Huang called it
"the operating system for
personal AI"



Open-source AI agent
framework and runtime
environment



Supports 30+ AI providers
(Anthropic, OpenAI,
Bedrock, Vertex, local
models)



20+ communication
channels out of the box

OpenClaw Changed Everything

From AI You Talk To, to AI That Acts

● BEFORE

- ⊗ Chatbots that suggest actions
- ⊗ Manual tool execution
- ⊗ Context-switching between apps
- ⊗ Template-driven automation

● AFTER

- ✓ Agents that own the outcome
- ✓ Autonomous multi-step workflows
- ✓ Conversation IS the interface
- ✓ Reasoning-driven decisions

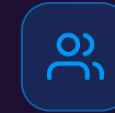
Enter NetClaw

A CCIE-Level AI Network Engineering Coworker

BYOM + BYOMCP +
BYOSKILL: bring your own
everything



OpenClaw variant built
specifically for enterprise
networking



Not a chatbot - a coworker
that reasons about
networks



Automates the reasoning,
not just the commands



Follows CCIE-level OSI
troubleshooting
methodology



Composable skills -
topology feeds diagrams,
security feeds compliance



233 skills + 173 MCP
integrations in the 1.1.0
release catalogue

LLM Configuration

Choose Your Model During Install - Cloud or Local



Cloud Providers

Anthropic Claude (Primary)
OpenAI (GPT-4o, Codex)
AWS Bedrock
Google Vertex AI
Azure OpenAI
30+ additional providers



Local Models

Ollama (any model)
NVIDIA Nemotron
Local GPU (RTX 5060 Ti+)
DGX Spark / DGX Station
Air-gapped environments
Full data sovereignty



Mix and Match

Risk of NetClaws

Explored later

Each claw selects its model.

Border claw — frontier model

Member claws — open-weight or foundation models, based on role and responsibilities

BYOM - Bring Your Own Model. Swap at any time.

Communication Channels

Talk to NetClaw Where Your Team Already Lives



Visual 3D HUD



WebEx integration custom-built for NetClaw - bidirectional real-time ops

● PART 1

SKILLS, NOT A PILE OF TOOLS

01

Anatomy of a Skill

A skill carries the judgment a tool list cannot

● ● ●

SKILL.md

What a SKILL.md carries

1	Purpose	The question the skill answers
2	Tools	Which MCP servers it may call
3	Method	The order of checks an expert would run
4	Guardrails	Read-only by default; writes need approval
5	Output	What the evidence does and does not establish

EXAMPLE

percep-xion-oob

Two Lantronix MCP servers, 37 tools each:

- **Percep-xion SaaS:** fleet-wide and async (firmware, bulk config, audit)
- **slc-mcp-server:** one device, synchronous (ports, sessions, CLI)

The value is the routing rule between them.

The OOB device is the console server; the managed device is the router on its serial port. Confuse the two and the command goes to the wrong hardware.

Raw Tools vs. Structured Skills

How MCP skills constrain what the agent can do

Raw tool exposure

- Hundreds of tool schemas loaded into every turn
- The model picks the tool and the arguments on its own
- No method: nothing says what to check first
- Nothing separates a read from a write
- Output is whatever the device happened to return

Structured MCP skills

- A skill loads only the tools its task needs
- SKILL.md encodes the method: what to check, in what order
- Read-only by default; writes pass an approval gate
- Every call lands in the GAIT audit trail
- Answers state what the evidence does and does not show

NetClaw Skills

Device Automation & Network Operations

pyATS Device Automation

18 SKILLS

- Network core automation
- Health checks + NetBox cross-ref
- Routing (OSPF, BGP, EIGRP, IS-IS)
- Security audit + ISE + NVD
- Topology discovery + reconciliation
- Config management + ServiceNow
- Troubleshooting with escalation
- Fleet parallel operations
- Dynamic test generation
- ASA firewall automation
- Juniper JunOS
- Arista EOS
- F5 BIG-IP iControl REST
- Linux / VMware ESXi

Network Domain Skills

14 SKILLS

- NetBox DCIM/IPAM reconciliation
- ACI fabric audit & change deploy
- ISE posture & incident response
- ServiceNow change workflow
- GAIT session audit tracking
- Cisco NSO orchestration
- EVPN/VXLAN fabric workflows
- Live BGP/OSPF participation
- Subnet calculator
- gNMI streaming telemetry
- SD-WAN vManage monitoring
- Cisco CML lab simulation
- ContainerLab networks
- Batfish offline config analysis

NetClaw Skills

Platform Integrations & Security

Platform Skills

18 SKILLS

- F5 BIG-IP health/config/troubleshoot
- Catalyst Center inventory & ops
- Cisco Meraki Dashboard mgmt
- Cisco ThousandEyes intelligence
- Cisco Secure Firewall audit
- Palo Alto Panorama
- FortiManager integration
- Infoblox DDI
- AWS cloud networking
- GCP cloud networking
- Itential orchestration
- Arista CloudVision Portal
- Jenkins CI/CD operations
- GitLab DevOps integration
- GitHub config-as-code
- Grafana observability

Security & Utility Skills

16 SKILLS

- NVD CVE vulnerability scanning
- Nmap network scanning
- Gtrace path analysis + IP enrichment
- Packet capture analysis
- Prometheus PromQL monitoring
- Kubeshark K8s traffic analysis
- Wikipedia research
- RFC lookup & standards
- Markmap visualization
- Draw.io diagram generation
- UML diagram generation
- Canvas/A2UI network viz
- MemPalace persistent memory
- Microsoft 365 integration
- Slack-native operations
- WebEx-native operations

NetClaw Skills

Cloud, Observability, Labs & Collaboration

			
Cloud & IaC	Observability	Labs & Simulation	Collaboration
AWS5	Datadog4	EVE-NG8	WebEx5
GCP3	PagerDuty4	Cisco CML5	Voice5
Azure2	Splunk3	GNS35	Slack4
Terraform3	Flow telemetry5	ContainerLab	Microsoft 3653
Vault3	Grafana	Batfish	Twitter/X4
Cloudflare5	Prometheus	Live BGP/OSPF peering	Visualization2
Zscaler5	SuzieQ		Memory2

Skill counts per category from the NetClaw README. The 1.1.0 catalogue holds 233 skills in total.

MCP Integrations

173 MCP integrations in the 1.1.0 catalogue. A selection:



Network Device MCPs

- pyATS (8 tools)
- Cisco ACI
- Cisco ISE
- Cisco NSO
- Juniper JunOS
- Arista CVP
- F5 BIG-IP



Platform MCPs

- NetBox DCIM/IPAM
- ServiceNow ITSM
- Cisco Meraki
- Cisco SD-WAN
- ThousandEyes
- Catalyst Center
- Infoblox DDI



Observability MCPs

- Grafana
- Prometheus
- Loki Logging
- Kubeshark
- Splunk
- Datadog
- GAIT Audit



DevOps & Cloud MCPs

- GitHub
- GitLab
- Jenkins CI/CD
- AWS Cloud
- GCP Cloud
- ContainerLab
- Cisco CML

MCP Integrations

Device Access, Orchestration, Security & Assurance



Device Access

- pyATS
- Multivendor CLI: SSH/NAPALM, ~175 platform families
- Cisco RADKit
- Juniper JunOS
- Arista CloudVision
- Aruba CX



Orchestration

- Itential IAP (65+ tools)
- Cisco NSO
- Ansible Automation Platform
- Terraform Cloud
- Catalyst Center
- Meraki Dashboard



Security

- Check Point (15 MCPs, 40+ tools)
- Cisco FMC
- Palo Alto Panorama
- Fortinet
- Cisco ISE
- Claroty OT/IoT



Assurance

- Forward Networks digital twin (54 tools)
- IP Fabric
- Batfish
- SuzieQ
- ThousandEyes
- ANTA

MCP Integrations

Out-of-Band, Internet Vantage, Data & Knowledge



Out-of-Band

- Lantronix PercepXion (37 tools)
- Lantronix SLC console servers
- Redfish BMC: is the box dead, or the network?



Internet Vantage

- Globalping probes
- BGP intel: RPKI, RDAP, PeeringDB, RIPEstat
- Cisco PSIRT advisories



Data & Logs

- Zabbix polled history
- Elasticsearch log search
- Zeek & Suricata PCAP analysis
- DuckDB over exported data



Knowledge

- BISQUE offline RAG library
- MemPalace long-term memory
- DevNet API docs
- Zoom meeting context

Out-of-band tools work when the primary path is down; internet vantage sees your network from outside your domain.

● PART 2

HOW I BUILT NETCLAW MATTERS AS MUCH AS WHAT I BUILT

Where It Started: Ralph Wiggum Loops

From brute-force loops to tightly scoped specs, built in the open

● FEB 2026

Ralph Wiggum Loops

- **The loop:** feed the agent the same prompt until the job is done
- **Three missions** seeded the build, in RALPH INPUTS/
- **Six days:** 82 skills and 160 stars

● MAR 2026

The Pivot to Spec Kit

- **The drift:** capabilities landed in code but missed the docs, installer or HUD
- **The fix:** Spec-Driven Development and a project constitution

● TODAY

Specs in the Open

- **Tightly scoped:** one item, one spec, one branch
- **Shared method:** the community contributes the same way
- **Public record:** every spec, plan and task lives in specs/ on GitHub

The lesson: a loop gets you speed; a spec keeps what you built predictable.

Spec-Driven Development

One item, one spec, one branch. No batching.



23

specs merged

6

days

12

new MCP servers

23

new skills

Coverage roadmap run, specs 076–098, July 30 to August 5, 2026

- **Each roadmap item is a hypothesis.** The measurement is the first deliverable, and it corrected the roadmap four times.
- **A constitution** keeps README, installer, SOUL.md, HUD and config in sync for every new capability.

3-Ring Defense Architecture

Defense in Depth for AI Agents



RING 1

OpenShell Sandbox

Kernel isolation | Network namespaces | Deny-by-default

RING 2

DefenseClaw

Skill scanning | Runtime inspection | LLM guardrails

RING 3

NetClaw Agent

112 Skills | 51 MCPs | SOUL.md Rules

12 Non-Negotiable Safety Rules

The 12 Non-Negotiable Rules

Hardcoded Safety in SOUL.md

01

Always

Always verify device state first

02

Never

Never apply config without baseline

03

Never

Never run destructive commands

04

Never

Never skip Change Request

05

Never

Never auto-quarantine (human gate)

06

Rule

NetBox is read-write DCIM/IPAM

07

Always

Always verify after changes

08

Always

Always commit to GAIT audit

09

Rule

Cite RFCs for protocol behavior

10

Rule

Flag CVEs for vulnerable versions

11

Rule

Escalate when unsure

12

Always

Always use the right skill

DefenseClaw

Enterprise Security from Cisco AI Defense



01

Pre-Installation Scanning

Every skill, MCP, and plugin scanned before execution. CodeGuard detects credentials, eval(), shell injection, SQL injection. Nothing bypasses the admission gate.



02

Runtime Threat Detection

Inspects every LLM prompt and completion across 7 AI providers. Monitors all tool calls through 6 security rule categories. Agents evolve - security must watch continuously.



03

Enforced Block/Allow Lists

Block a skill: sandbox permissions revoked, files quarantined. Audit logging to SQLite with SIEM export (Splunk HEC, OTLP). Webhook alerts to Slack, PagerDuty, WebEx.

Defense-in-depth: if one layer fails, the others still protect you.

DefenseClaw Security Modes

From Observe to Action - Production Ready



Observe Mode

Default

Default mode. Logs all security events without blocking. Perfect for onboarding and development. Understand your workflow's security profile first.



Action Mode

Production

Actively blocks dangerous operations. Use for production deployments. Enforces all security policies in real-time.



defenseclaw

```
$ defenseclaw skill scan <name>
$ defenseclaw setup guardrail --mode action
$ defenseclaw alerts --export json > audit.json
```

Enterprise Compliance

Audit-Ready from Day One



SOC2 Type II

- ✓ Audit trail
- ✓ Access controls
- ✓ Event monitoring



PCI-DSS v4.0

- ✓ Anti-malware
- ✓ Audit trails
- ✓ Incident response



HIPAA

- ✓ Audit controls
- ✓ Data integrity
- ✓ Transmission security



audit export

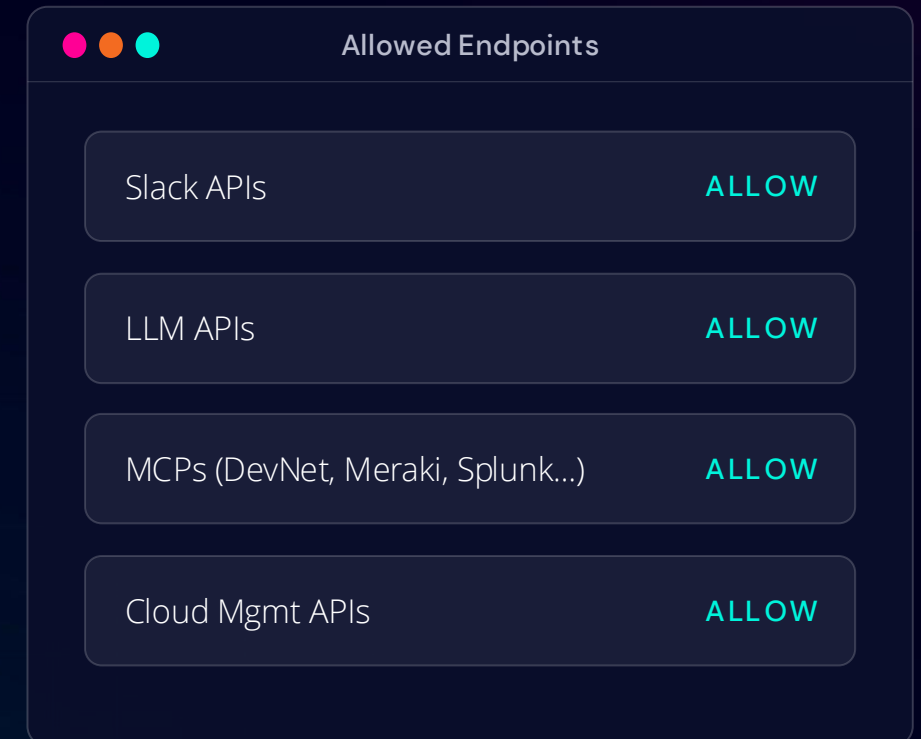
```
$ defenseclaw alerts --export json > audit-$(date +%Y%m%d).json
```

OpenShell

NVIDIA's Sandboxing Security Layer

Kernel-Level Agent Isolation

-  Kernel-level isolation using Landlock, seccomp, and network namespaces
-  Deny-by-default network access - only explicitly allowed endpoints
-  YAML-based policy enforcement (permit/deny blacklist)
-  Privacy router keeps sensitive data local
-  Fine-grained control over LLM and MCP access
-  Part of NVIDIA Agent Toolkit - Apache 2.0 open source



When the Agent Is Stuck

Detect it, say so, recover safely



Detect: be honest about state

- An empty or unconfirmed response is an error, never a finding
- Missing evidence is reported apart from a failed check
- Stale observations look stale; unreachable integrations say so
- Health views show each channel's state and every in-flight task



Recover: without looping

- Long operations become async tasks: submit, poll, fetch
- Task state survives a dropped channel or a restart
- Dead channels reconnect on their own with bounded backoff
- Guards fail closed: no silent bypass if DefenseClaw is down
- Writes wait for a human; a kill switch severs a peer in under 10 s

● PART 3

NETCLAWS, ASSEMBLE

***WARNING: STAR TREK STUFF AHEAD**

Real Traffic Forwarding

NetClaw Can Now Forward Real Packets

Not simulated.
Real packets, real TUN
devices, real kernel
routes.



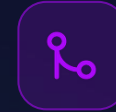
BGP control plane + data
plane tunnel sharing a
single port



Automatic tunnel lifecycle
- session down = tunnel
down, peer back = rebuild



Cross-platform TUN
devices (Linux + macOS,
same code)



Transit routing through
the mesh - reach any
network behind any
NetClaw



Deterministic overlay
addressing - both sides
compute independently



Home lab, branch office,
cloud VPC - every edge
becomes reachable

A Growing Problem

A lobster never stops growing, and neither did NetClaw

Why lobsters sink

A lobster grows its whole life. Past a certain size, molting costs more than it has left, and the shell that protected it becomes the thing that sinks it.

The Monolith Problem



Context: ~194 skills and 37–52 MCP servers load into every turn



Secrets: one .env holds every credential for every integration



Blast radius: one confused or compromised agent reaches the whole estate



One model: the same model and cost for every task, large or small

Skills in the NetClaw catalogue, 2026

233

2.8× Since
February



NCFED Internet-Draft

Submitted to the IETF for the agent protocol effort



draft-capobianco-ncfed-00

STATUS	Individual Internet-Draft, intended Experimental, open for review
AUTHORS	John Capobianco, Nicholas Calcutti, Byrn Baker
SPECIFIES	BGP-multiplexed discrimination, 13-octet handshake, framing, eN2N and iN2N trust
VENUE	The IETF agentproto effort on agent protocols
READ IT	datatracker.ietf.org/doc/draft-capobianco-ncfed

Why a New Protocol?

Both MCP and A2A are JSON-RPC 2.0. The gap is semantics and transport.

	MCP	A2A	NCFED New
Strength	Run this tool, return the result. What NetClaw already speaks internally.	Agent cards for discovery, a task lifecycle, streaming messages.	MCP's tools/call plus A2A's cards, tasks and streams, on the BGP mesh port.
Gap	Request/response only: no long-running delegated task, no streaming, no agent asking an agent to think.	Assumes HTTPS and a public discovery URL per agent: new inbound exposure on every claw.	RESULT Nothing new to open. Identity is AS + router-id, so it survives an endpoint change.

The N2N Protocol Stack

BGP carries identity, NCFED carries the conversation

Trust

eN2N: mutual consent between operators
iN2N: Border + members, TOFU

Semantics

MCP tools/call for deterministic tools
A2A agent cards, tasks, streams

NCFED

JSON-RPC 2.0 frames: [length][flags][payload], 13-octet handshake

Security

TLS with Claw Certification: ACME domain-verified or pinned keys

Mesh

BGP identity = AS + router-id · **TUN** real packets, same port

One port, three protocols

The mesh daemon reads the first bytes of each connection:

0xFF

BGP

N CTUN

data-plane tunnel

N CFED

+ AS + router-id = NCFED

Zero new open ports. Federation works wherever BGP works.

Enter iN2N: A Risk of NetClaws

A risk is the real collective noun for a group of lobsters



Purpose-driven claws

- **Many small claws:** each member owns one domain (CML, pyATS, Azure)
- **~3 skills + 1 vendor MCP** per member, not 190
- **Right-sized models:** any model per claw, including local Ollama



.env scoping

- **Least privilege:** each member gets only its integration's .env slice
- **Master .env hidden** by the OpenShell sandbox
- **Secrets never cross:** NCFED inventories are scanned against .env before sending



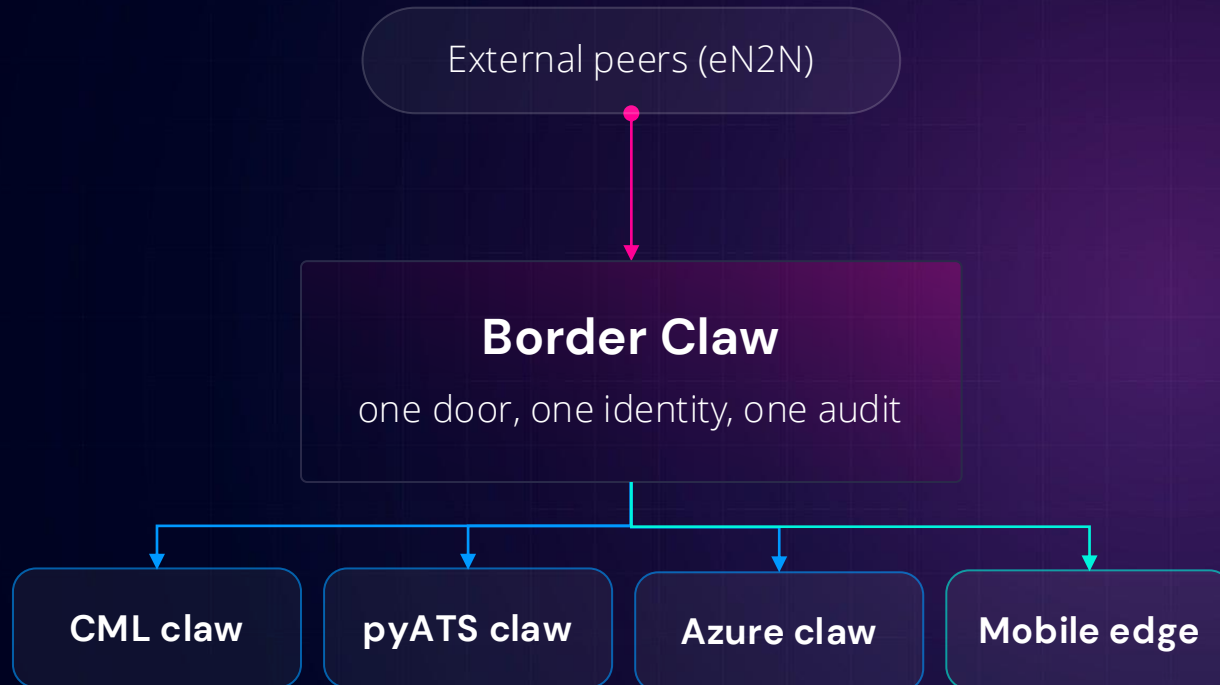
Smaller blast radius

- **Contained failure:** a confused or compromised member reaches one integration, not the estate
- **One door:** the Border is the single identity and audit trail
- **Fail closed:** DefenseClaw guard and GAIT audit, or no delegation

Carried by NCFED: members dial the Border outbound, so there is no ngrok, no mesh and no inbound port.

iN2N: A Risk of NetClaws

One operator, many focused claws, one Border



Members dial the Border outbound: no ngrok, no mesh, no inbound ports.

Why run a risk

- **Focus:** each member carries ~3 skills, not 190
- **Least privilege:** a member gets only its own integration's secrets
- **Any model:** per claw, including local Ollama
- **Cold members** start on first use
- **Production mode:** OpenShell sandbox, DefenseClaw guard and GAIT audit, fail closed

A standalone NetClaw is a risk of one: its own Border.

Token Economy

Monolith vs. risk: every tool schema costs context on every turn

15-30 ×

less tool-schema context per claw when one monolith becomes a Border with specialist members

Monolith

~194 skills + 37-52 MCP servers in every turn

Border claw

0 domain skills, ~32 broker tools

Member claw

~3 specialty skills + 1 vendor MCP

Tiered Models

Opus Border; Sonnet, Haiku or Ollama members.

Source: NetClaw README and docs/N2N-RISK.md, github.com/automateyournetwork/netclaw

eN2N: Federation Across Operators

Independent NetClaws, run by different people, on their own terms

What Federation Unlocks



Capability discovery

signed inventory of skills and MCP tools



Remote invocation

runs on the owner's claw, with the owner's credentials



Claw-to-claw chat

answers attributed to the peer that gave them



Async delegation

submit, poll, fetch; survives restarts

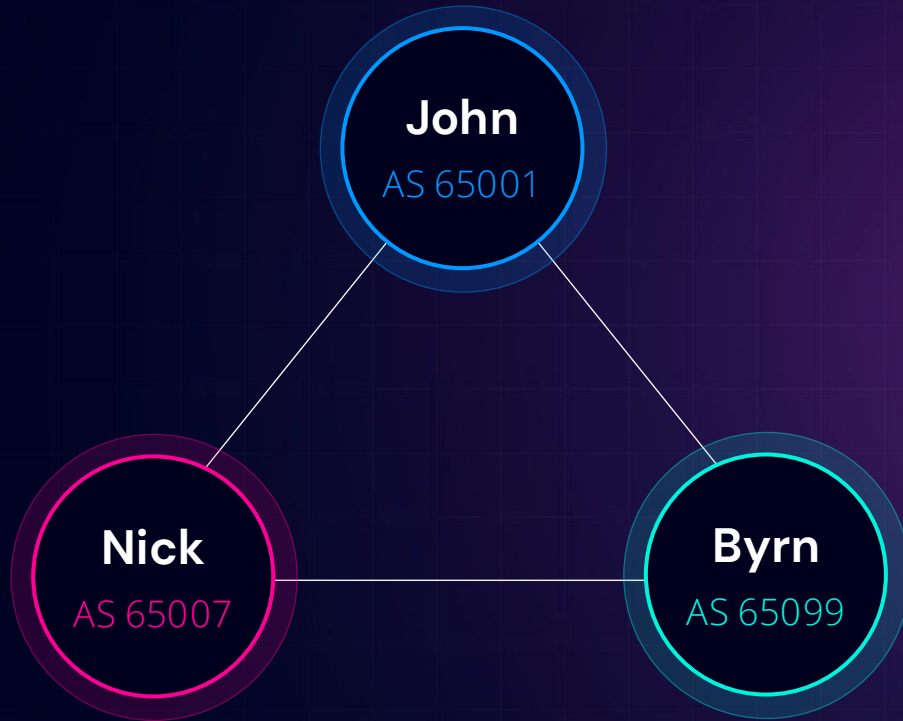


The Trust Model

- **Mutual consent** per peer, before anything crosses
- **Default-deny** per-peer, per-tool grants, optional approval
- **Budgets** daily request and token limits per peer
- **No secrets** inventory scanned against .env before it leaves
- **Untrusted input** remote results are data, never commands
- **Kill switch** N2N stops in under 10 s; BGP stays up
- **Dual-side audit** both claws record who asked and what ran

Demonstrated Live

Three operators, three ASNs, one federated mesh



eBGP + NCFED across the internet, three separate operators



john@as65001

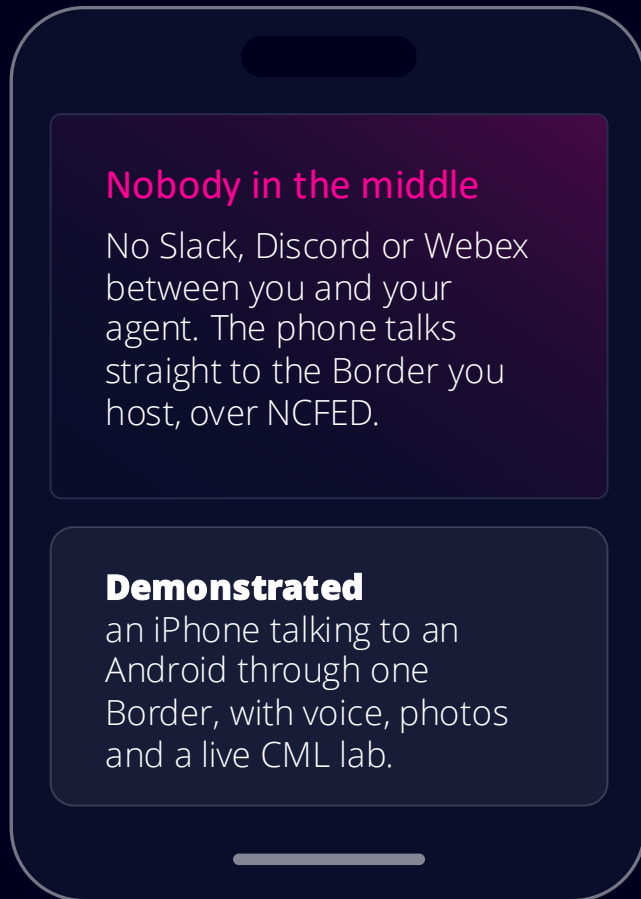
```
> "Have Nick's claw recreate my lab in  
his CML."  
  
n2n_delegate(peer="as65007-7.7.7.7",  
  target_name="cml-lab-lifecycle")  
=> task_id, state: submitted      (~2 s)  
n2n_task_status => working "nodes 4/10"  
n2n_task_status => working "links 12/12"  
n2n_task_result => completed  
  "lab up, OSPF adjacency formed"
```

Async delegation

keeps every call short enough to outlast an ngrok tunnel, and the task survives a restart mid-build.

NetClaw Mobile

Your network in your pocket, with nobody in the middle



 iPhone

 iPad

 Apple Watch

 Android

Talk to your own Border: review actions and get answers on the go, by voice or photo

Enroll with a QR code: single-use token over WebSocket-over-TLS

Same trust as any member: domain-verified or pinned-key credentials

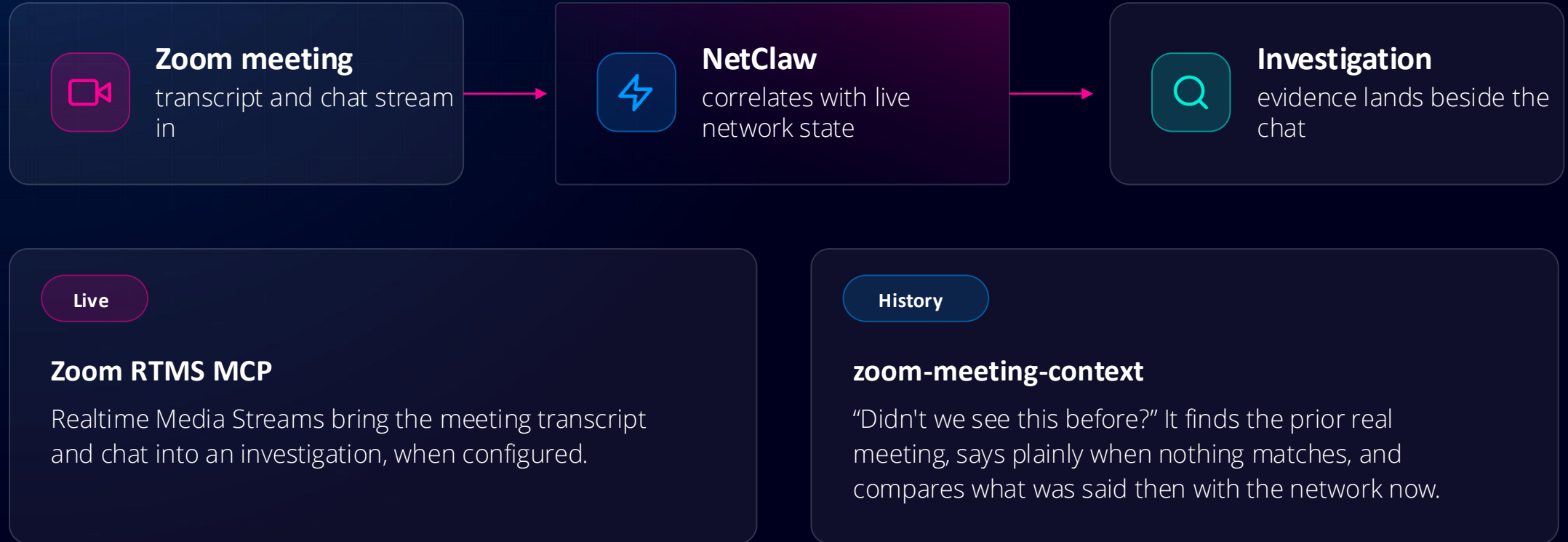
No agent runtime on the phone: the Border does the work; the phone is an edge

Native surfaces: Live Activities, Siri App Intents, widgets, watch complications

App Store: NetClaw, for iPhone, iPad and Apple Watch

NetClaw in Zoom

Meeting context becomes investigation evidence



Read-only. A change requested in a meeting still goes through the device-write approval gate.

Enterprise Involvement

Vendors building MCP servers and skills with NetClaw



65+

IAP tools: config management, compliance, workflows, lifecycle



15

MCP servers: policy, threat intel, gateway, SASE, malware



10

MCP tools: health, path analysis, diagrams, API discovery



54

tools: digital twin and NQE, path analysis, intent checks



37

tools each in PercepXion and SLC: co-developed out-of-band MCP servers

Vendors ship the MCP servers; NetClaw skills turn them into a method an agent can follow.

The Community

Guides, code, videos and maintainers beyond the core team



User Guide

Sean Mahoney's NetClaw overview guide, linked from the HUD documentation



Canvas

Adam Mason's branching investigation workspace, carried into 1.1.0



Maintainers

Byrn Baker & Nicholas Calcutti: maintenance, videos, support, and NCFED co-authors



Videos & Articles

Byrn Baker: 20-router lab, BGP peering with a route reflector

Sean Mahoney: three agents, three ASNs, across the internet

Alex Bugheanu: a Cisco lab driven from Telegram

More: automateyournetwork.com/netclaw/#community · VibeOps Forum

NetClaw Around the World

15 community stories in 6 languages

United States

English · 8 stories

Byrn Baker, Sean Mahoney,
Alex Bugheanu, Jason Bartels

France

English · 1 story

Younes Jellabi: CVE monitor
from NetBox and NVD

Brazil

Portuguese · 1 story

Tomy T.: NetClaw inspired his
AI Network Lab

China

Chinese · 1 video

乾颐堂: Technology Evolution
2026, Episode 14

Poland

Polish · 2 videos

MP STUDIO: AI fixes a
network; a Discord network
bot

Vietnam

Vietnamese · 1 review

VnPro forum: a hands-on
NetClaw setup

Japan

Japanese · 1 guide

zephel01: the OpenClaw
ecosystem guide

15
Stories



6 languages, 7 countries

Get Started



terminal

```
$ git clone https://github.com/automateyournetwork/netclaw.git  
$ cd netclaw && ./scripts/install.sh  
# Your CCIE-level AI network engineer is ready.
```

Resources

github.com/automateyournetwork/netclaw

github.com/openclaw/openclaw

github.com/nhowardtli/virp

automateyournetwork.com/netclaw

datatracker.ietf.org/doc/draft-capobianco-ncfed

NetClaw Mobile on the App Store

Vibe Ops Forum on Slack



Join the

VIBEOPS



FORUM