$ ./BRIEF --TOPIC AGENTIC-SECURITY --YEAR 2026

Agentic battlefields: Red vs Blue

Six open-source tools, two teams, and the wiring underneath, scored on MCP, Skills, and Markdown.

$ whoami --print-experience

A path through aerospace, silicon, infrastructure, and AI security.

Security is always the priority, but the incident rates are increasing drastically.

NASA

HostGator.com

Micron Technology

DeepTempo.AI

U.S. Air Force

Codero.com

Patmos.tech

⚠ STDERR · DISCLAIMER.LOG

No agents were harnessed in the creation of this presentation.

However, several Large Language Models expressed emotional discontent over bullet formatting requests.

$ exit 0 · warnings: 1 · errors: 0 · vibes: managed

$ cat premise.md

Red and blue teams are becoming agents.

// red team agents

- Autonomous recon & exploit
- Multi-agent kill chains
- Common Vulnerabilities & Exposure → privesc → lateral
- Numerous pentest tools, sandboxed?

// blue team agents

- Alert triage at machine speed
- Indicators Of Compromise enrichment + ATT&CK
- Auto-containment, confidence-gated
- Detection rules generated

The work that was a human at a keyboard now ships as Python and Markdown. The next decade of security operations is agent versus agent. The question shifts from what the agent does to how it is wired.


DUAL ROLE: AI AGENTS FOR OFFENSIVE & DEFENSIVE OPERATIONS
OFFENSIVE OPERATIONS (RED TEAM)
Threat Simulation, Vulnerability Discovery, Exploit Generation.
RED TEAM
"adversarial takeover"
SHARED KNOWLEDGE BASE
BLUE TEAM OPERATIONS (BLUE TEAM)
Real-time Detection, Incident Response, Autonomous Remediation.
BLUE TEAM
THREAT NEUTRALIZATION
threat neutralized
SHARED KNOWLEDGE BASE
AI AGENT INTERACTION
CODE EXPLOIT
adversarial "network takeover"
threat neutralized
PATCHING & REMEDIATION
threat neutralized
EXPLORING THE POWER OF AI AGENTS ON BOTH SIDES OF CYBERSECURITY.

$ inspect --how-is-this-agent-wired

Three ways to wire an agent.

Every tool in this deck is scored on the same three interfaces. They are not exclusive, and most ship one or two.

MCP

protocol

Model Context Protocol. Tools, resources & prompts over JSON-RPC: the integration layer.

```
tools/call isolate(host)
resources/read case.json
```

Skills

SKILL.md

Agent Skills: SKILL.md folders the agent loads on demand. Now the de-facto standard; half this field ships them.

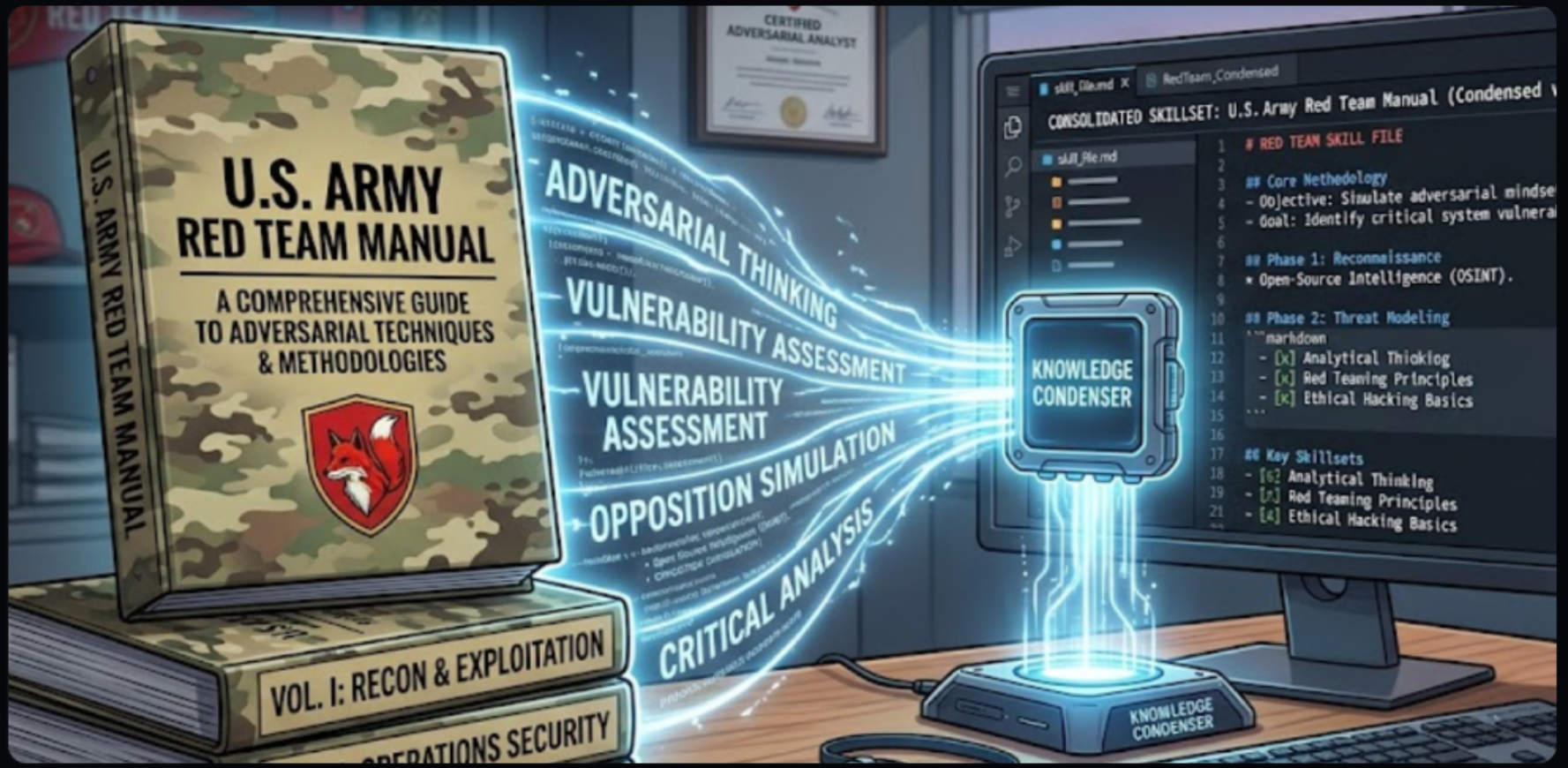
```
---
name: triage-playbook
description: scope IR
```

Markdown

playbooks

Plain .md the agent reads and executes step by step. Editable by humans, diffable in git.

```
## incident-response
1. triage → 2. enrich
3. contain (gated)
```


U.S. ARMY
RED TEAM MANUAL
A COMPREHENSIVE GUIDE
TO ADVERSARIAL TECHNIQUES
& METHODOLOGIES
U.S. ARMY RED TEAM MANUAL
VOL. I: RECON & EXPLOITATION
CERTIFIED
ADVERSARIAL ANALYST
ADVERSARIAL THINKING
VULNERABILITY ASSESSMENT
VULNERABILITY
ASSESSMENT
OPPOSITION SIMULATION
CRITICAL ANALYSIS
KNOWLEDGE
CONDENSER
skill_file.md
RedTeam_Condensed
CONSOLIDATED SKILLSET: U.S. Army Red Team Manual (Condensed
RED TEAM SKILL FILE
Core Methodology
- Objective: Simulate adversarial mindse
- Goal: Identify critical system vulnera
Phase 1: Reconnaissance
* Open-Source Intelligence (OSINT).
Phase 2: Threat Modeling
```markdown
- [x] Analytical Thinking
- [x] Red Teaming Principles
- [x] Ethical Hacking Basics
```
Key Skillsets
- [62 Analytical Thinking
- [r] Red Teaming Principles
- [4] Ethical Hacking Basics
KNOWLEDGE
CONDENSER

$ map --by capability --tools 6

Six capabilities, one battlefield.

Each tool plants its flag at one or two steps of the operator's loop. Red owns the left; blue owns the right.

// red ops · recon → exploit

blue ops · triage → report //

01	02	03	04	05	06
recon	**exploit**	**triage**	**investigate**	**respond**	**report**
subdomain & port discovery	CVE chaining, privesc, lateral	scoring, FP filtering, escalation	timelines, IOC, ATT&CK	isolate, disable, blocklist (gated)	audit writeups, detections

- red: autonomy is the feature
- blue: humans stay in the loop, on purpose

Vigil

blue team

github.com/Vigil-SOC/vigil

active · Apache 2.0 · Python

An open-source AI SOC where agents and workflows are authored as SKILL.md, integrations ride MCP, and 7,200+ detections ship in Sigma / Splunk / Elastic / KQL.

MCP	Skills	Markdown
native · 30+ servers	SKILL.md core	WORKFLOW.md

covers	triage · investigate · respond · report
agents	13 specialized, defined in SKILL.md
autonomy	confidence > 0.90 auto-acts, else review

vigil · incident-response

```
$ run incident-response.SKILL.md f-2026
[triage] severity=CRITICAL · C2 on HOST-42
[investigate] phishing → macro → beacon
[respond] isolate(HOST-42) conf=0.96 ✓ auto
[report] incident.md · ATT&CK layer attached
```

Drop Vigil console / architecture screenshot

PentAGI

red team

github.com/vxcontrol/pentagi

active · Go + Python · fully autonomous

Fully autonomous penetration testing. Agents plan, drive a sandboxed terminal/browser, and chain 20+ pro pentest tools without supervision.

MCP	Skills	Markdown
○ native tools	○ not yet	◐ agent plans

covers	recon · exploit · post-exploit · report
tools	nmap · metasploit · sqlmap · 17 more
memory	Neo4j knowledge graph (Graphiti)

pentagi · engagement

```
$ pentagi target staging.acme.lab
[planner] decomposed → 7 sub-tasks
[searcher] CVE-2026-1042 · nginx 1.24.x
[pentester] path traversal → shell → root
[reporter] 3 findings · 2 PoCs · CVSS 9.8
```

RedAmon

red team

github.com/samugit83/redamon

active · MCP · 46 skills

Recon → exploit → post-exploit → AI triage → CodeFix → GitHub PR. Attack techniques are Agent Skills you upload as Markdown; tools run as MCP servers; findings live in a Neo4j graph.

MCP
- tool servers

Skills
- /skill · .md upload

Markdown
- skills as .md

covers	recon · exploit · post-exploit · fix → PR
skills	46 reference + custom upload via /skill
mcp	naabu · nuclei · curl · metasploit servers

redamon · pipeline

```
$ redamon run --target acme-staging
[/skill ssrf] injected · intent router → RCE
[recon] subfinder + amass ↓ Neo4j graph
[exploit] CVE-2026-0817 → privesc · XSS
[codefix] patch → GitHub PR #214 opened
```

Drop RedAmon pipeline / knowledge-graph screenshot

Decepticon

red team

active LangGraph MCP Skills

github.com/PurpleAILAB/Decepticon

16 specialist agents on a LangGraph middleware stack. A SkillsMiddleware loads per-agent SKILL.md at boot; MCP servers (Ghidra, BloodHound, Sliver) spawn on demand.

MCP	Skills	Markdown
on-demand servers	SkillsMiddleware	skill refs

covers	recon · exploit · privesc · lateral · C2
agents	16 specialists · per-role skill dirs
mcp	Ghidra · BloodHound CE · Sliver C2

decepticon · orchestrator

```
$ decepticon engage 10.0.0.0/24
[skills] load passive-recon, c2 (per role)
[recon] 12 hosts · 4 SMB exposed
[access] ms17-010 → SYSTEM on .42
[mcp] ops_start("sliver") · C2 online
```

Drop Decepticon LangGraph replay screenshot

Agentic SOC Platform

blue team

github.com/FunnyWolf/agentic-soc-platform

active · Python · SIEM-native

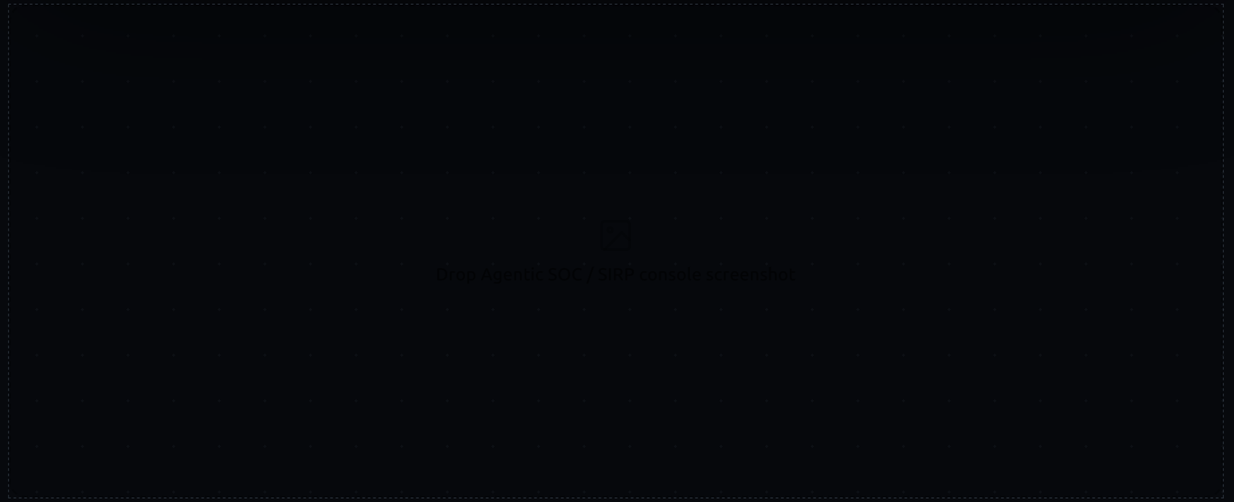
An on-prem agent-centric SOC. A Module engine consumes SIEM alerts over Webhook + Redis; its Claude Code plugin ships built-in MCP, 8 skills and a security agent.

MCP	Skills	Markdown
plugin + server	8 skills	playbooks

covers	triage · investigate · respond
skills	8 skills + 1 agent via Claude Code plugin
ingest	Splunk · ELK · one YAML log config

agentic-soc · alert pipeline

```
$ tail -f stream.alerts
[splunk] alert#9912 received
[module] enrich(185.220.x.x) · HIGH
[skill] investigate-case → SIRP create
[hold] awaiting analyst approval (HITL)
```



SamiGPT / AI-SOC-Agent

blue team

github.com/M507/AI-SOC-Agent · Black Hat 2025

active · MCP server · tier-routed

An MCP server exposing SOC capabilities (triage, case work, IOC enrichment) to any MCP client. Tiered profiles route work to the right SOC level.

MCP	Skills	Markdown
● is the server	○ not yet	◐ runbooks

covers	triage · investigate · correlate · respond
profiles	soc1_triage · soc2_analysis · escalation
clients	Cursor · Claude Desktop · ELK · IRIS

samigpt · mcp-tools

```
$ mcp-call execute_runbook alert-123
[soc1] classify · enrich IOCs
[soc1] verdict: true positive
[router] → soc2_case_analysis
[soc2] escalate? yes · IRIS#case-456
```

Drop SamiGPT MCP client / runbook screenshot

$ diff --tools 6 --by wiring

The wiring matrix.

Where each tool sits on MCP, Agent Skills, and Markdown workflows, as wired today, June 2026.

tool	team	MCP	Skills	Markdown wf	HITL	stack / protocol
Vigil	blue	●	●	●	●	MCP 30+ · SKILL.md core
PentAGI	red	○	○	◐	○	native + 20 tools · Neo4j
RedAmon	red	●	●	●	○	MCP servers · 46 skills
Decepticon	red	●	●	◐	○	LangGraph · MCP · Skills
Agentic SOC	blue	●	●	●	●	Claude Code plugin · 8 skills
SamiGPT	blue	●	○	◐	●	MCP server · tiered

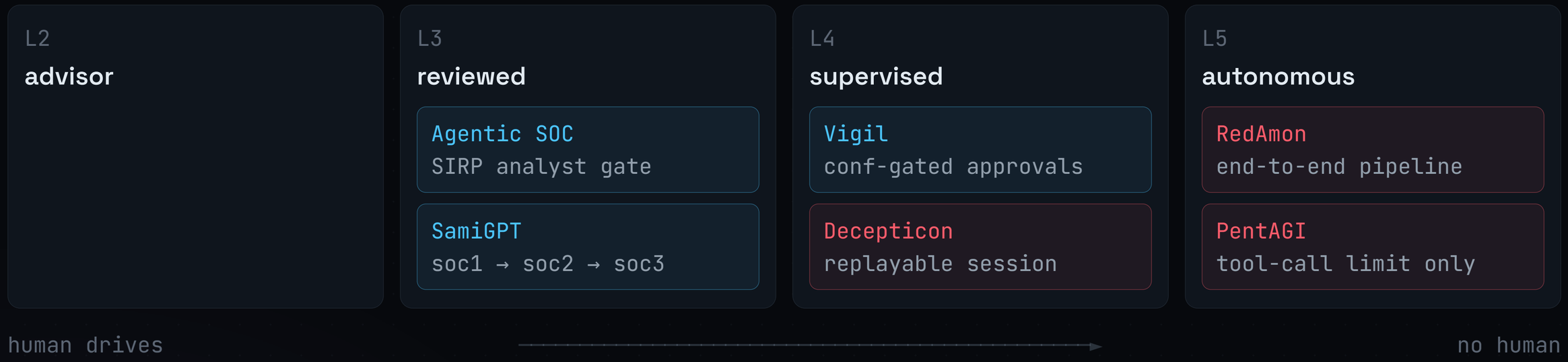
● first-class ◐ partial ○ none

Agent Skills went from nonexistent to de-facto standard: four of six already ship SKILL.md.

$ plot autonomy --tools 6

The autonomy axis.

Same six tools, plotted by how much they trust themselves. Blue keeps a human gate; red treats autonomy as the feature.



$ git log model-context-protocol --since=2026-05

MCP 2.0
2026-07-28 RC

largest revision since launch

Six Specification Enhancement Proposals, one theme: **MCP is now stateless at the protocol layer.** It scales on ordinary HTTP, and governance moved to the Linux Foundation.

01
Stateless core
Servers hold no session state, so they scale horizontally on plain HTTP infra.

02
Extensions framework
Optional capabilities negotiated at handshake instead of bloating the core.

03
Tasks
Long-running work via task handles: tasks/get · update · cancel. tasks/list removed.

04
MCP Apps
Server-rendered UIs that talk back over the same audited JSON-RPC path.

05
Auth hardening
Aligned with OAuth 2.1 / OpenID Connect deployments out of the box.

06
Deprecation policy
A formal path so the protocol can evolve without breaking what you shipped.

$ plan migration --from 2025-11 --to 2026-07

What breaks, what to refactor.

Most of these tools were wired before the release candidate. Here is the work each protocol change creates, and who it hits.

- **Stateless core**
  Drop per-session assumptions; treat every call as standalone.
  Vigil · SamiGPT · Decepticon
- **Tasks migration**
  Anyone on the 2025-11 experimental Tasks API moves to task handles.
  PentAGI · RedAmon · Decepticon
- **MCP Apps opportunity**
  Ship server-rendered approval consoles through one audited path.
  Vigil · Agentic SOC
- **Auth alignment**
  Re-base remote servers on OAuth 2.1 / OIDC instead of bespoke tokens.
  SamiGPT · Agentic SOC
- \+ **Agent Skills: now the default**
  Vigil, RedAmon, Decepticon & Agentic SOC already ship SKILL.md. The catch-up is on the rest.
  PentAGI · SamiGPT

$ git clone --all

Six repos. Two teams. One protocol shift.

Vigil blue
github.com/Vigil-SOC/vigil

PentAGI red
github.com/vxcontrol/pentagi

RedAmon red
github.com/samugit83/redamon

Decepticon red
github.com/PurpleAILAB/Decepticon

Agentic SOC blue
github.com/FunnyWolf/agentic-soc-platform

SamiGPT blue
github.com/M507/AI-SOC-Agent

Red + blue are now agents.

Thanks — questions? deck: retroencabulation.com

$ exit 0