THE CYBER TOOLCHAIN NO. 005

Head

THE WEEK'S TOP PICKS

PUBLISHED JULY 31, 2026 · FRIDAYS

// SHARE THIS WEEK ▽ Top 5 ▽ Top Pick

The best tooling updates that shipped this week.

// HOW THESE PICKS ARE MADE

Every feature release from the 874 tools on our watchlist goes into the daily newsletter. Once a week we read the whole field side by side and pick the few updates worth your attention, on two questions: **how deep and complete is the single best capability in the release**, and **how much it changes what you can actually do**.

A tool is judged on everything it shipped that week, so a project that releases daily gets credit for the sum — and still only takes one slot. We would rather run a short list than a padded one.

// THE PICKS

01	**Wiz Defend**	Runtime Security
02	**Intruder**	Attack Surface & Exposure Management
03	**OpenAI Codex CLI**	AI Coding Agents
04	**tcpreplay**	Network Analysis & Monitoring
05	**ConductorOne**	Identity & Access Management

01 Wiz Defend

1 RELEASE · 2026-07-29

Runtime Security

Wiz Defend is a cloud detection and response platform that identifies and helps remediate real-time threats across cloud environments.

// WHY IT MADE THE LIST

DEPTH 5/5 IMPACT 5/5

Red Agent, a continuous autonomous AI pentester, is now GA—reasoning through business logic to find exploitable flaws in APIs and custom/vibe-coded apps and delivering proofs of execution with reproduction steps. New Web API Crawler, logic-driven API DAST, and Secrets Blast Radius modules also validate exposed secrets against real third-party SaaS APIs.

> Wiz Defend brought its Red Agent AI pentester to general availability with three new modules for API discovery, logic-driven exploitation, and secrets validation, while also launching an AI Threat Defense partnership with Google and introducing a SecOps-focused Blue Agent.

└──▷ WHAT SHIPPED · 6 FEATURES most completely described first ? what's the number?

01 API DAST Attacker module for logic-driven exploits NEW

65

New API DAST Attacker module continuously finds and validates logic-driven exploitable risks — including OWASP API Top 10 flaws, authorization bypasses, and logic flaws — adapting its strategy in real time and delivering proofs of execution with reproduction steps.

— Names OWASP category and gives reproduction-step output as a starting point

2026-07-29-6c541523

THINNER COVERAGE BELOW

02 Red Agent AI pentester reaches GA NEW

58

Red Agent is now generally available: a continuous, autonomous AI pentester that reasons through business logic to uncover unknown vulnerabilities across custom-built software, vibe-coded applications, and APIs. It automatically ingests context from Wiz sources — API schemas, cloud resources, Runtime Sensor, code repositories, and network layers — without manual setup.

— Explains autonomy and context sources but no setup steps 2026-07-29-6c541523

03 Secrets Blast Radius module validates exposed secrets NEW

55

New Secrets Blast Radius module validates secrets exposed in public websites, APIs, and code repositories by executing safe checks against third-party SaaS provider APIs to confirm exploitability and determine the precise scope of permissions held.

— Describes verification mechanism but no interface to run it 2026-07-29-6c541523

04 **Web API Crawler module for shadow API discovery** NEW 50

New Web API Crawler module uses client-side code analysis to map API endpoints, extract and analyze API specifications, and surface hidden or forgotten shadow APIs.

— Names the mechanism but no way to invoke it directly 2026-07-29-6c541523

05 **AI Threat Defense partnership with Google** NEW 30

AI Threat Defense launched in partnership with Google, combining Wiz, Mandiant, and Google DeepMind capabilities.

— Names partners but no detail on what the combined capability does 2026-07-29-6c541523

06 **Blue Agent SecOps AI analyst** NEW 30

Blue Agent introduced: a SecOps AI analyst that automates threat detection, triage, and containment workflows.

— States purpose only, no mechanism or scope given 2026-07-29-6c541523

GOOD PICK? 👍 👎 GOOD REASON? 👍 👎

02 Intruder

1 RELEASE · 2026-07-28

Attack Surface & Exposure Management

Intruder is a cloud-based vulnerability scanner that continuously monitors and identifies security weaknesses in internet-exposed systems, helping businesses reduce their attack surface.

// WHY IT MADE THE LIST

DEPTH 4/5 IMPACT 4/5

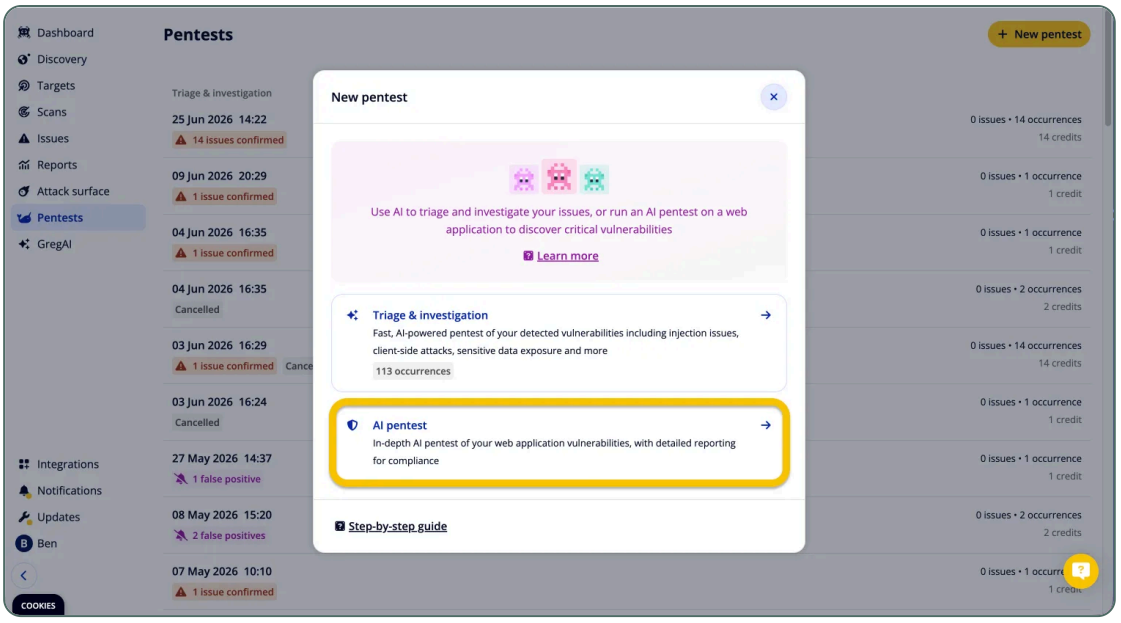
AI Pentesting delivers full web app penetration tests on demand, producing a PDF report with executive summary, severity ratings, reproduction steps, and remediation. It adds whitebox testing via code-repo integration and detects access-control, business-logic, and auth-bypass flaws that require multi-step chaining and are missed by static or dynamic scanning.

> Intruder launched AI Pentesting, an on-demand web application penetration testing capability that combines whitebox code analysis with logic-flaw detection and produces full PDF reports.

└──▷ WHAT SHIPPED · 1 FEATURE most completely described first ? what's the number?

01 **AI Pentesting for web applications** NEW 80

A new AI Pentesting capability runs full web app penetration tests on demand, producing a PDF report with executive summary, severity ratings, reproduction steps, and remediation guidance. It supports whitebox testing via code repository integration for deeper analysis beyond surface-level scanning, and detects access control, business logic, and auth bypass vulnerabilities that require multi-step chaining and are missed by static or purely dynamic scanning. Practitioners can define custom entry points, credentials, and application context to scope tests to their real environment; to start one, go to 'AI Pentesting' in the portal's left nav, click 'Start new pentest', then choose the 'AI Pentest' option.



Start an AI-driven pentest against your web app directly from the Intruder portal.

> 📍 In the portal, go to AI Pentesting in the left nav, click 'Start new pentest', then choose the 'AI Pentest' option.

— Mechanism and report contents named; exact UI path given but no API/config surface.

`AI Pentesting - expert-level web app pen…`

GOOD PICK? 👍 👎 GOOD REASON? 👍 👎

03 OpenAI Codex CLI

11 RELEASES · 2026-07-23 → 2026-07-30

AI Coding Agents

Lightweight coding agent that runs in your terminal

// WHY IT MADE THE LIST

DEPTH 4/5 IMPACT 4/5

Named and pinnable sessions with thread forking, Agent Plugins manifests, and WebSocket Code Mode land in v0.146.0, giving practitioners persistent multi-thread agent workflows. The latest alpha completes MCP 2026-07-28 client support for the newest Model Context Protocol spec and exposes cached MCP tools before server startup to cut latency.

> Codex CLI's biggest push this window was MCP protocol modernization — full 2026-07-28 discovery and MCP 2026 client support plus a string of connection, caching and reliability upgrades — alongside a new Agent Plugins manifest ecosystem, remote Code Mode host support, and expanded thread management (naming, pinning, forking).

└──▷ WHAT SHIPPED · 28 FEATURES most completely described first ? what's the number?

01 **Remote Code Mode host support** NEW 95

The app-server can connect to a remote Code Mode host over WebSocket via the new `--code-mode-host` flag (e.g. `wss://...`). The `features.code_mode_host` config table gains a `disable_in_process_fallback` option to prevent silent fallback to the embedded V8 host when the standalone host fails to start, and Responses Lite turn metadata gains a `code_mode_tool_names` field exposing normalized code-mode identifier-to-ToolName mappings for each turn.

Prevent silent fallback to embedded V8 so failures in the standalone code-mode host surface as tool output rather than being swallowed.

```toml
# In your Codex config file:
[features.code_mode_host]
disable_in_process_fallback = true
```

Connect the app-server to a remote code-mode host over a secure WebSocket instead of the default in-process host.

```
$ codex-app-server --code-mode-host wss://example.com/code-mode
--listen ws://127.0.0.1:9000
```

— exact flag, config key, and runnable examples given `rust-v0.146.0` `rust-v0.146.0-alpha.10` `rust-v0.146.0-alpha.6`

02 **Thread naming, pinning, and side conversations** NEW 90

New sessions can be named via `/new` or `/clear`; users can pin important threads and switch between side conversations without closing them, and side conversations stay open when switching threads. Thread responses now include an `isPinned` field, with pin/unpin support via the `thread/metadata/update` endpoint and an `isPinned` filter on `thread/list`, plus cursor-based pagination.

— names commands, endpoints, and fields precisely `rust-v0.146.0` `rust-v0.146.0-alpha.6` `rust-v0.146.0-alpha.5`

03 **Agent Plugins manifest support and marketplace expansion** NEW 88

Codex recognizes Agent Plugins 1.0 manifests — root `plugin.json` files mapping metadata, `skills/`, and `mcp.json` into Codex plugin manifests — and applies Codex-specific apps, hooks, and interface settings from the inline `com.openai` extension, with `.codex-plugin/plugin.json` as a fallback overlay. Plugins can be published to a workspace, surfaced via `canPublishToWorkspace` capability metadata in `plugin/share/save` responses. Additional plugin marketplaces are supported for Amazon Bedrock (API plugin marketplace) and Claude Code (inferred bundled marketplace); curated plugin recommendations are routed by authentication mode and prepared concurrently with MCP recommendations; remote plugin catalogs are cached by scope; and trusted plugin script attribution tracks which plugin scripts originated command executions.

— enumerates every named manifest field and endpoint `rust-v0.146.0` `rust-v0.146.0-alpha.14` `rust-v0.146.0-alpha.8` `rust-v0.146.0-alpha.7` `rust-v0.146.0-alpha.6` `rust-v0.146.0-alpha.5`

04 **Remote exec server control and networking** NEW 85

Remote exec servers gain a `--exit-on-stdin-close` flag (and `CODEX_EXEC_SERVER_EXIT_ON_STDIN_CLOSE` environment variable) for graceful shutdown with telemetry flush when the parent process closes stdin, network policy callbacks for remote exec environments, exec-server HTTP and WebSocket traffic routed through the configured proxy policy, and streaming body support in route-aware HTTP requests.

Keep a remote exec server's lifetime tied to the spawning process — useful in CI or orchestration where you want the server to exit cleanly when the parent dies rather than becoming an orphan.

```
$ codex exec-server --remote --exit-on-stdin-close
```

— exact flag, env var, and runnable command given `rust-v0.147.0-alpha.2` `rust-v0.147.0-alpha.1` `rust-v0.146.0-alpha.6`

05 **Thread forking with paginated history** NEW 80

Threads can be forked with paginated history, including temporary forks that do not appear in thread listings. The `thread/fork` endpoint supports an `excludeTurns: true` parameter for ephemeral forks that preserve selected history and a fork preview without creating a persistent rollout.

— names endpoint and parameter, mechanism explained `rust-v0.146.0` `rust-v0.146.0-alpha.8` `rust-v0.146.0-alpha.7`

06 **MCP connection handling, caching, and reliability** IMPROVED 78

MCP tooling gained a string of connection and performance improvements: cached MCP tools are exposed to inference before the MCP server finishes initializing; MCP OAuth discovery and authentication-status requests are routed through each server's configured runtime HTTP client for proxy-aware auth; MCP tool name prefixes can be omitted on a per-server basis; MCP connections are reused across runtime refreshes, with closed connections replaced during reconciliation; MCP runtime updates are prewarmed in the background; MCP runtimes refresh automatically when session authentication changes; all configured MCP servers reconnect on an explicit refresh trigger; MCP HTTP requests now carry a default User-Agent header for auditability; and refreshed Apps tools are preserved across MCP runtime updates.

— names every mechanism but no single runnable command `rust-v0.147.0-alpha.2` `rust-v0.146.0-alpha.13` `rust-v0.146.0-alpha.8` `rust-v0.146.0-alpha.7` `rust-v0.146.0-alpha.6` `rust-v0.146.0-alpha.5`

07 **Standalone web search for custom model providers** NEW 70

Custom Responses API model providers can opt into the standalone `web.run` web search tool via the new `supports_standalone_web_search` model-provider setting; when enabled, standalone web search requests are routed through the custom provider's own endpoint and authentication, extending web-search capability beyond first-party models.

– names setting and tool, no config snippet shown `rust-v0.146.0` `rust-v0.146.0-alpha.3.1` `rust-v0.146.0-alpha.5`

---

08 **Plugin eligibility metadata in app-server** NEW 60

Exposes plugin eligibility metadata (`disabledReason` , `eligiblePlanTypes`) in app-server plugin summaries, surfacing plan-gating and admin-disabled status.

– names exact fields, no client usage shown `rust-v0.147.0-alpha.1`

---

09 **Parent turn tracking for nested agent spawns** NEW 60

Adds `parent_turn_id` tracking across nested Codex agent spawns, delegated sessions, and follow-up tasks for improved request provenance.

– names field and scope, no query example `rust-v0.147.0-alpha.1`

---

10 **Per-turn network approval scoping** IMPROVED 60

Scopes pending network approvals to a turn and execution, coalesces duplicate requests within one execution, and assigns unique per-execution identifiers to user approval requests and permission hooks.

– explains mechanism, no config surface named `rust-v0.146.0-alpha.10`

---

11 **Customizable wait_for_environment tool descriptions** NEW 60

Allows hosts to customize the model-visible `wait_for_environment` tool and `environment_id` descriptions via `WaitForEnvironmentToolConfig` .

– names exact config type, no usage snippet `rust-v0.146.0-alpha.7`

THINNER COVERAGE BELOW

12 **New ent26 enterprise and Business ProLite plan tiers** NEW 55

Supports the `ent26` enterprise plan tier, including authentication, rate-limit payloads, cloud-config eligibility, and usage-limit guidance, and adds support for self-serve Business ProLite accounts.

– names tiers and scope but no signup flow `rust-v0.147.0-alpha.2` `rust-v0.146.0-alpha.8`

---

13 **Windows process and path handling improvements** IMPROVED 55

On Windows, non-TTY processes are now terminated on interrupt, with Ctrl-C routed through the termination callback for pipe-backed processes, and Windows namespace paths in path URIs are normalized.

– describes mechanism, platform-specific fix `rust-v0.147.0-alpha.2`

rust-v0.146.0-alpha.14

14 **Configurable disabling of update_plan and multi-agent wait tools**

NEW

55

Operators can disable the `update_plan` tool via configuration, and can disable the multi-agent wait tool, giving control over multi-agent coordination primitives.

– names both tools, no exact config key given rust-v0.146.0-alpha.6

rust-v0.146.0-alpha.5

15 **Cloud-managed sandbox profiles for codex sandbox** NEW

50

Loads cloud-managed profiles for `codex sandbox`, allowing centrally managed sandbox permission profiles and requirements to be sourced from cloud config bundles rather than local config.

– names the command but no config schema shown rust-v0.147.0-alpha.2

rust-v0.146.0-alpha.14

16 **MCP 2026 protocol support** NEW

45

Adds `MCP 2026-07-28` discovery support and completes MCP 2026 client support, enabling Codex to use the latest Model Context Protocol spec.

– names spec version but no usage mechanism given rust-v0.147.0-alpha.2

17 **Model-owned token budget defaults** NEW

45

Model catalog entries can now supply default token-budget settings that apply automatically when no explicit token-budget configuration is set.

– explains behaviour but no config key named rust-v0.147.0-alpha.2

rust-v0.146.0-alpha.13

18 **Configurable developer instructions for v2 subagents** NEW

45

Adds configurable developer instructions for v2 subagents, allowing per-subagent instruction customization in multi-agent workflows.

– describes scope, no exact config surface rust-v0.147.0-alpha.2 rust-v0.146.0-alpha.14

19 **Executor-provided skills discovery and resource access** NEW

45

Discovers executor-provided skills and securely reads their associated resources, including explicitly selected skills, exposing them through skill tools.

– describes capability, no interface details `rust-v0.146.0` `rust-v0.146.0-alpha.7`

20 **Thread item replay and ownership enforcement** IMPROVED

40

Supports incremental replay of updated thread items and enforces writer ownership for thread archive and deletion operations.

– describes two changes with no mechanism detail `rust-v0.146.0-alpha.6`

21 **Turn report tagging with model and effort level** NEW

35

Tags turn reports with the selected turn's model and effort level.

– names fields tagged but no report format shown `rust-v0.147.0-alpha.2`

22 **Hooks stdout preservation for fast-exiting hooks** IMPROVED

35

Preserves output from hooks that exit before reading stdin, so fast-exiting hooks no longer lose stdout.

– bug-fix style description, no reproduction steps `rust-v0.146.0-alpha.7`

23 **App-server exposes skill icons and Browser Use requirements** NEW

35

Exposes remote skill icon URLs through the app server, and exposes Browser Use requirements through the app server.

– two thin bullets with no further mechanism `rust-v0.146.0-alpha.6`

24 **Concurrent thread title loading at startup** IMPROVED 30

Loads thread titles concurrently during session startup, improving startup responsiveness.

– states benefit but no metric or mechanism `rust-v0.147.0-alpha.2`

25 **Persisted thread sections in session history** NEW 25

Adds persisted sections for organizing threads in session history.

– bare description of the feature only `rust-v0.147.0-alpha.2`

26 **WebRTC sideband joins routed to Realtime API** IMPROVED

25

Routes WebRTC sideband joins to the Realtime API.

– single-line description, no mechanism `rust-v0.147.0-alpha.2`

27 **Plaintext collaboration tool messages** NEW

25

Supports plaintext collaboration tool messages.

— one line, no further detail supplied `rust-v0.147.0-alpha.2`

28 **Guardian V2 feature flag registered** NEW

15

Registers the Guardian V2 feature flag.

— bare flag name, no behaviour described `rust-v0.146.0-alpha.6`

GOOD PICK? 👍 👎 GOOD REASON? 👍 👎

04 tcpreplay

1 RELEASE · 2026-07-28

Network Analysis & Monitoring

Edit and replay captured network traffic at arbitrary speeds — a suite of pcap tools for *NIX and Windows.

// WHY IT MADE THE LIST

DEPTH 4/5 IMPACT 5/5

New AF_XDP and io_uring injection backends push replay to ~941 Mbps line rate on 1GigE hardware—a 23× improvement—while a new --loss flag simulates packet loss and the replay engine now installs as a standalone libtcpreplay static C library. Raw L3 injection for WireGuard/tun interfaces is also added.

> tcpreplay 4.6.0 adds kernel-bypass and io_uring injection backends for much higher throughput, a raw-socket path for replaying onto L3-only interfaces, packet-loss simulation, a standalone static library, and a CMake build option.

└──▷ WHAT SHIPPED · 6 FEATURES most completely described first ? what's the number?

01 AF_XDP kernel-bypass injection backend NEW

89

New `--xdp` injection backend reaches ~941 Mbps (305k pps) line rate on 1GigE hardware, a 23× improvement over the previous implementation. `--xdp-queue` binds the `AF_XDP` socket to a specific adapter queue instead of always using queue 0, `--xdp-no-fallback` fails hard rather than silently falling back to the default injector (useful so a silent change doesn't invalidate benchmark results), and `--xdp-batch-size` tunes `AF_XDP` batch depth for very high-speed links (100GigE+) with small packets.

Saturate a 1GigE link using the `AF_XDP` kernel-bypass backend; fall back automatically if the adapter lacks native XDP support.

```
$ tcpreplay --xdp -i eth0 capture.pcap
```

Benchmark `AF_XDP` on a specific queue without silent fallback masking results — fail loudly if XDP can't be used.

```
$ tcpreplay --xdp --xdp-no-fallback --xdp-queue 2 -i eth0
capture.pcap
```

— Names every XDP flag with concrete throughput numbers and runnable commands. v4.6.0

02 io_uring batched injection backend NEW

70

New `--io-uring` injection backend batches sends 64 deep via Linux io_uring, reducing per-packet syscall count 64× for roughly 11% less CPU per packet.

– Mechanism and numbers given but no usage example. v4.6.0

---

03 **--raw socket injection for L3-only interfaces** NEW 70

New `--raw` flag adds a `PF_INET` / `SOCK_RAW` injection path that routes packets through the kernel IP stack (routing, netfilter); it is IPv4-only and rebuilds L2 framing. This also enables replaying onto raw IP (L3-only) interfaces such as WireGuard and tun, which have no L2 header.

– Names the flag, socket type, and interface targets but no example command. v4.6.0

---

04 **--loss packet-loss simulation flag** NEW 63

New `--loss` flag simulates random packet loss at a configurable percentage from 0-100%, enabling testing against non-ideal network conditions.

Simulate a lossy link by randomly dropping 10% of packets during replay to stress-test application resilience.

```
$ tcpreplay --loss 10 -i eth0 capture.pcap
```

– Exact flag and runnable example, but limited mechanism detail. v4.6.0

---

05 **CMake build system option** NEW 63

Adds a CMake build system as an alternative to autotools, invoked via `cmake -B build && cmake --build build`, with full feature parity.

– Exact build commands given, though scope of parity is unelaborated. v4.6.0

THINNER COVERAGE BELOW

---

06 **Standalone libtcpreplay static library** NEW 32

The replay engine is now installed as `libtcpreplay`, a standalone static C library.

– Names the library but gives no usage or API detail. v4.6.0

GOOD PICK? 👍 👎 GOOD REASON? 👍 👎

05 ConductorOne

2 RELEASES · 2026-07-25 → 2026-07-27

Identity & Access Management

ConductorOne is an identity governance and access management platform that automates user access reviews, provisioning, and deprovisioning across enterprise applications.

// WHY IT MADE THE LIST

DEPTH 4/5 IMPACT 4/5

Assign an approval step in a request or review policy to the C1 AI agent, which evaluates each request against custom guidance with every decision attributed to a system identity and recorded as a reviewable, audit-logged conversation. Per-user authentication for external MCP servers, group/role access reviews, and self-service scoped API keys round out the release.

> ConductorOne's headline addition this window lets policy authors delegate request and review steps to a configurable C1 AI agent with fully audited decisions, alongside access reviews that can now scope to groups and roles, queued provisioning for pre-created entitlements, per-user authentication for MCP servers, and scoped personal API keys.

└──▷ WHAT SHIPPED · 14 FEATURES most completely described first ? what's the number?

01 AI agent steps in request and review policies NEW 80

Policy authors can assign a workflow step in a request or review policy to the C1 AI agent, configuring agent mode (comment-only, full control, or policy-step changes), failure action (reassign or skip), and free-text approval instructions. The agent evaluates each request using built-in guidance plus the custom instructions, and every evaluation is attributed to the C1 System agent identity, labeled as an AI-agent action in the audit log, and recorded as a reviewable conversation under AI > Conversations.

Delegate a policy step to the C1 AI agent so it automatically evaluates access requests and logs each decision for audit — no human reviewer needed for routine approvals.

📍 In the console, go to Policies › [your policy] › Edit, add or select a workflow step, choose Assign to agent, enter any custom instructions, and save. Audit log entries for that step will appear under the C1 System agent identity with the AI-agent action label.

Set up an AI agent step in a policy to auto-evaluate low-risk access requests and reassign to super admins if it can't decide.

📍 In the console, go to Policies › [your policy] › Edit, add a new step, select Agent as the step type, set Agent mode to 'Full control', set Agent failure action to

'Reassign to super admins', and enter Agent instructions such as 'Approve if the requester is in the contractors group and the resource is tagged low-risk; otherwise deny.'

— Full mechanism and audit trail described, with step-by-step UI path.

July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

02 **Access review campaign scope and scheduling options** NEW

70

Access review campaigns can now review grants held by groups or roles in addition to users, using a new 'Grant types' selector (Users only, Resources only, or Users and resources) on the Scope tab. Campaign templates also support custom recurrence intervals (e.g., every two days, every three months) and configurable campaign durations, beyond the previous fixed presets.

Run a quarterly access review that covers group and role grants, not just individual user grants, to satisfy broader compliance requirements.

📍 In the console, go to Access Reviews › Create Campaign, open the Scope tab, find the Grant types selector, choose 'Users and resources', set a custom recurrence interval (e.g., every three months), configure the campaign duration, and launch.

Include groups and roles in a quarterly access review campaign to catch over-provisioned shared identities alongside user grants.

📍 In the console, go to Access Reviews › Campaigns › Create Campaign, open the Scope tab, and set the Grant types selector to 'Users and resources' before launching.

— Combines scope selector and scheduling options with exact UI steps.

July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

03 **Queued requests for pre-provisioned entitlements** IMPROVED

65

Requests for entitlements created ahead of a connector sync (e.g., via Terraform) now queue and wait rather than fail, display a status reason, resume automatically after the next sync, and fall back to manual provisioning after 24 hours.

— Clear before/after behavior with a concrete 24-hour fallback. July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

04 **MCP server admin controls: enablement, OAuth checks, tool prefix**

NEW

65

Tool enablement settings (require enablement or auto-enable) can now be configured per MCP server, independently of the tenant-wide AI governance default. Admins can register MCP servers that fail relaxable OAuth discovery checks (e.g., issuer mismatch, cross-host authorization server) by reviewing and acknowledging flagged findings. Hosted MCP servers without an admin-set tool prefix now default to the prefix declared by the connector itself, keeping tool names recognizable to AI clients.

— Names three distinct MCP admin controls with specific check examples.

July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

05 **Scoped personal API keys** NEW

65

Any user can now restrict a personal API key to a narrower role (e.g., read-only) using the 'Limit Scopes' selector at key creation, without requiring Super Admin access.

Scope a personal API key to read-only to limit blast radius when integrating with a third-party reporting tool.

> 📍 In the console, go to Settings › Personal API Keys › Create API Key, open the Limit Scopes selector, choose the read-only role, and generate the key.

Scope a personal API key to read-only to safely hand credentials to a reporting script without granting full account access.

> 📍 In the console, go to your profile › API Keys › Create API Key, then open the Limit Scopes selector and choose the read-only role before saving.

— Exact selector name plus step-by-step creation path. July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

06 **Per-user authentication for external MCP servers** NEW

60

Admins can now require per-user authentication (OAuth2, bearer token, custom header, or basic auth) for external MCP servers instead of sharing a single admin credential.

— Names four auth methods and the before/after credential model.

July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

THINNER COVERAGE BELOW

07 **Entitlements matching in request catalog search** NEW

55

Request catalog search now includes an 'Entitlements matching' section showing entitlements that indirectly grant access to a searched resource (e.g., a GitHub team

whose repo permissions include the target repo), with a 'View access graph' link showing the full grant chain.

— Names the UI section and link but no worked example given. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

08 **Data lineage view for AI assistant reports** NEW 55

C1 AI assistant reports now include a 'View lineage' button showing the exact queries behind each number, verified against the execution log, and copyable for audit workpapers.

— Names the button and verification mechanism, no example given.

July 24, 2026 (2026-07-27) July 24, 2026 (2026-07-25)

09 **Readable AWS IAM policy viewer in access reviews** NEW 55

IAM policy documents on AWS resources now render as a readable, collapsible JSON tree via a 'View policy' button on resource-level access reviews and in task details.

— Names exact button and location but no deeper mechanism. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

10 **Retry failed provisioning action on access tasks** NEW 50

A new 'Retry failed provisioning' action on access tasks lets admins retry connector provisioning in place without canceling the task and losing collected approvals.

— Names the exact action but no UI navigation given. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

11 **Multi-chart grids and composed reports in C1 AI** IMPROVED 45

C1 AI now renders multiple requested charts as an aligned grid with individual export and source links, and can respond with composed reports combining stat cards, a trend chart, and a data table in one response.

— Describes new output format without a runnable example. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

12 **Groups requestable via catalog and access profiles** NEW 45

C1 groups can now be added to a request catalog or access profile, making group membership requestable through the standard approval, duration, and revocation flows.

— States the change but gives no UI path or example. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

13 **Generate action in CEL editor** NEW 45

A new 'Generate' action in the CEL editor converts a plain-English description directly into a CEL condition expression in one step, without requiring iterative AI chat.

— Names the exact button but no config key or command. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

14 **Combined step-up MFA and comment approval flow** IMPROVED 35

Step-up (MFA) approvals that also require a comment now prompt for identity-provider verification in a single flow and complete once verified.

— Describes behavior change but no concrete flags or steps. July 24, 2026 (2026-07-27)

July 24, 2026 (2026-07-25)

GOOD PICK? 👍 👎 GOOD REASON? 👍 👎

// END OF TRANSMISSION

The Cyber Toolchain · This Week's Highlights · July 31, 2026
The full firehose — every feature release, every day — is in the newsletter.

▷ Subscribe