THE CYBER TOOLCHAIN

NO. 101

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED OCTOBER 10, 2026 · EVERY DAY

The daily firehose — everything the toolchain shipped today, already filtered.

▸ // HOW THIS ISSUE IS MADE SHOW

VIEW

ISSUE VIEW

scroll



SCROLL | EXPAND IN PLACE | PREVIEW PANE

INFORMATION DENSITY

everything



BRIEFED | EVERYTHING

Every tool in the issue, in full.

all 39 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 5 tools

▾ // THE BRIEF HIDE

What stands out across today's releases, grouped by what it lets you do. Every tool named links to its entry below.

Three unrelated things stand out. Dalfox can now take an OpenAPI or Postman file as the target list, so XSS testing covers documented oints a crawler never reaches. Detect It Easy added PE heuristics

vim ? help

that name RAT, stealer and ransomware families statically. Vigolium put hard budget ceilings on agentic scans, so an autonomous run cannot bill forever.

PROTECT

Point an XSS scanner at your API spec instead of guessing routes

Dalfox ingests OpenAPI 3.x/Swagger 2.0 and Postman Collection v2.1 with `--base-url` to retarget the documented host, which replaces hand-building URL lists for endpoints no crawler links to; a `--min-confidence` floor applies the same way from CLI, config, REST and MCP, so a noisy run can be tuned once rather than per interface. Vigolium's verify/mark lifecycle and CI report formats let a finding be dispositioned once and stay that way across runs.

Dalfox · noir · vigolium

DETECT

Get a family name off a sample before you open a disassembler

Detect It Easy's PE heuristic engine ships static coverage for dozens of RAT, stealer, ransomware and virus families plus known-plaintext detection of hidden payloads, which means triage can sort a dropped binary into a bucket without detonation. Plaso adds structured event classes for SSH, cron and sudo syslog activity, so the lateral-movement and persistence lines in a timeline parse into fields instead of free text.

detect-it-easy · plaso

GOVERN

Cap what an autonomous scanner or MCP server is allowed to do on its own

Vigolium enforces hard budget limits with auth and preflight gates on agentic scans; OpenOSINT makes its MCP server passive by default and puts every active investigation behind an explicit flag, with per-session caps on tool-call volume. Both close the same gap: an agent that decides its own next step was previously bounded only by whoever noticed. Runlayer extends Shadow AI discovery to Office add-ins and browser extensions, the two places unsanctioned AI usually enters unseen.

vigolium

OFFENSIVE SECURITY

◆ Reconnaissance & OSINT

OpenOSINT

Feedback ⓘ v2.33.0 NOTES ↗ RANK

AI-powered OSINT agent with interactive REPL, MCP server, and CLI. 20 tools.

> OpenOSINT v2.33.0 flips its MCP server to passive-by-default, gating all active investigation behind a new flag, and adds session-level controls for tool-call volume and noise.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Passive mode default for MCP server** BREAKING 80

The MCP server now defaults to passive mode, exposing 20 passive tools without `--allow-active`; all 25 tools require `--allow-active` to enable active investigation. This is a breaking change — active tools that previously ran by default now need the `--allow-active` flag explicitly set.

— Names exact flag and tool counts; clear migration step. v2.33.0

THINNER COVERAGE BELOW

02 **Tool-call cap per session** NEW 25

Adds a tool-call cap to limit the number of tool invocations allowed per investigation session.

— No limit value, flag or config key given. v2.33.0

03 **Per-tool noise labels** NEW 25

Introduces per-tool noise labels so practitioners can identify which tools generate high-noise output.

— No label scheme or surface to view them specified. v2.33.0

└──▷ **BREAKING ON UPGRADE**

! The MCP server now defaults to passive mode — active tools that previously ran by default now require `--allow-active` to be enabled.

WAS THIS USEFUL? 👍 👎

SensePost gowitness

Feedback ⓘ 3.2.1 NOTES ↗ RANK

gowitness - a golang, web screenshot utility using Chrome Headless

> gowitness 3.2.1 adds native ARM64 support to its official Docker images.

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

01 **ARM64 Docker image targets** IMPROVED 40

The official Docker images now include `arm64` targets, enabling native container deployments on ARM64 hosts.

— Names the exact target but no further mechanism or usage details 3.2.1

WAS THIS USEFUL? 👍 👎

DETECTION & RESPONSE

◆ Threat Intelligence

MISP

Feedback ⓘ v2.5.50 NOTES ↗ RANK

MISP (core software) - Open Source Threat Intelligence and Sharing Platform

> MISP 2.5.50 hardens sensitive configuration settings to CLI-only access, migrates several legacy subsystems (discussions, signature allowedlists, event templates) to their modern replacements, and adds a new object-relation usage statistics operation alongside a phishunt.io feed and Overmind UI refinements.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 CLI-only config hardening for sensitive settings

BREAKING

80

Adds `MISP.redis_host`, `Plugin.ZeroMQ_redis_host`, and `SimpleBackgroundJobs.redis_host` settings as CLI-only, and restricts `download_attachments_on_load` to CLI-only configuration — these can no longer be changed through the web interface or API, reducing risk from a compromised admin session.

— Names exact config keys and the access restriction mechanism. v2.5.50

02 Signature allowedlists migrated to warninglist

BREAKING

70

Migrates legacy signature allowedlists into an enabled custom regular-expression warninglist covering all attribute types. Filtering now applies only when `enforceWarninglist` is enabled — allowedlist filtering no longer unconditionally removes matching values from `restSearch`, TAXII push, OpenIOC, and Bro exports.

— Names the setting and affected export formats with clear before/after. v2.5.50

03 Legacy templates replaced with event templates

BREAKING

65

Replaces the legacy template system with event templates, converting organisation-authored templates to active event templates. The legacy ability to populate an existing event from a template is removed; event templates now only create new events. Legacy discussion, template, and allowedlist routes are removed, so integrations using those routes must be updated.

— Explains migration mechanism and removed capability, but no route paths given. v2.5.50

04 **Event discussions migrated to analyst notes** BREAKING

65

Migrates legacy event discussions (Thread/Post) to analyst notes, preserving author organisation, timestamps, and visibility information. The legacy Thread/Post discussion workflow trigger is removed; workflows that relied on it are logged during migration and will no longer fire.

— Describes mechanism and what breaks, moderate specificity. v2.5.50

05 **Object relation usage statistics operation** NEW

60

Adds `objectRelationUsage` statistics operation reporting how frequently relations are populated across MISP objects, grouped by the latest version of each object template.

— Names the operation and its grouping behaviour. v2.5.50

THINNER COVERAGE BELOW

06 **Overmind UI improvements for attributes and reports** NEW

55

Adds bulk actions in the Overmind event attribute index, warninglist match indicators next to attributes in Overmind views, and PDF downloads from event report quick actions in Overmind.

— Lists three concrete UI additions but no mechanism detail. v2.5.50

07 **phishunt.io phishing URL feed** NEW

40

Adds the phishunt.io feed of active phishing URLs as a new built-in feed.

— Names the feed but gives no further detail. v2.5.50

└──▷ **BREAKING ON UPGRADE**

! The `MISP.redis_host`, `Plugin.ZeroMQ_redis_host`, `SimpleBackgroundJobs.redis_host`, and `download_attachments_on_load` settings can no longer be changed via the web interface or API — they must be managed through server configuration or CLI.

! The legacy Thread/Post discussion workflow trigger is removed; workflows that relied on it are logged during migration and will no longer fire.

! Signature allowedlist filtering no longer unconditionally removes matching values from `restSearch`, TAXII push, OpenIOC, and Bro exports — exclusion now applies only when `enforceWarninglist` is enabled.

! The legacy ability to populate an existing event from a template is removed; event templates now only create new events.

! Legacy discussion, template, and allowedlist routes are removed; integrations using those routes must be updated.

WAS THIS USEFUL? 👍 👎

AGENT

◆ MCP Servers & Tooling

Runlayer

Feedback ⓘ 1.28.1158 (2026-10-10) BLOG ↗ RANK

Runlayer is an MCP gateway that enforces identity-mapped permissions and detects threats across the agents and MCP servers employees use.

> Runlayer 1.28.1158 adds autoscaling for deployed MCP servers, long-lived API keys for agent accounts, signed encryption for support-access diagnostics, and expands Shadow AI detection to Office add-ins and browser extensions, alongside improvements to Slack bot-filtering, audit logging, subagent context inheritance, and Assistant model configuration.

└──▷ WHAT SHIPPED · 9 FEATURES most completely described first ? what's the number?

01 Signed encryption for support-access diagnostics BREAKING

95

Adds `DEBUG_OPS_REQUEST_SIGNING_JWKS` (Kubernetes Secret) and `debug_ops_request_signing_jwks` Terraform input (ECS) to supply trusted public signing keys for signed, encrypted support-access diagnostic requests; missing or unusable keys disable diagnostic access entirely. ECS deployments must supply these keys via Terraform and Kubernetes deployments must set the Secret in the `runlayer-app` deployment before upgrading, or diagnostic support access is disabled.

Provide signed-support-access signing keys to a Kubernetes deployment so diagnostic requests remain available after upgrading.

yaml

```yaml
apiVersion: v1
kind: Secret
metadata:
  name: runlayer-app
stringData:
  DEBUG_OPS_REQUEST_SIGNING_JWKS: '<jwks-json>'
```

— Exact env var, Terraform input, Secret name, and migration requirement given.

1.28.1158 (2026-10-10)

02 Long-lived API keys for agent accounts NEW

85

Adds long-lived API keys for agent accounts, creatable from the Manage API keys menu on the agent account's detail page, usable on the MCP gateway in place of the client credentials flow; unavailable when `Require DPoP`, Strict mode, or Enforce audience is on.

Create a long-lived API key for an agent account to authenticate against the MCP gateway without the client credentials flow.

> 📍 In the Runlayer console, go to the agent account's detail page → Manage API keys → Create key. Copy the returned key and use it as a bearer token when calling the MCP gateway.

— Exact UI steps and interacting policy flags given. `1.28.1158 (2026-10-10)`

---

03 **Shadow AI detection for Office add-ins and browser extensions**

NEW

80

AI Watch Detect now inventories Office add-ins (Excel, Word, PowerPoint, Outlook, OneNote) on Windows and macOS, surfacing AI ones — including Claude for Excel/Word/PowerPoint/Outlook and the ChatGPT add-in — on the Shadow AI page with publisher, version, host application, store origin, and sideload status. It also inventories browser extensions across Chrome, Edge, Brave, Chromium, and Firefox profiles, surfacing AI ones — including Claude in Chrome, ChatGPT browser control, Perplexity, Grammarly, Browser MCP, Playwright MCP Bridge, and meeting recorders — with store, enabled state, installation method, and sideload status. The backend must be upgraded before unrecognized add-ins and extensions become visible.

— Names all surfaced apps and fields but only a UI page path. `1.28.1158 (2026-10-10)`

---

04 **Autoscaling for deployed MCP servers** NEW

75

Adds `infrastructure.scaling` to the infrastructure module for scaling deployed MCP servers at a fixed replica count or autoscaled on CPU and network use, with each MCP session pinned to one replica.

— Names config key and scaling behavior but no numeric limits. `1.28.1158 (2026-10-10)`

---

05 **Bot-scoped Slack channel monitors** IMPROVED

60

Adds bot-filtering to Slack channel monitors: with Respond to messages from bots on, monitors can be scoped to specific bots or apps selected by name from recently-posting bots or by message link, instead of requiring Slack app IDs.

— Names the toggle and selection methods, no config file reference.

`1.28.1158 (2026-10-10)`

THINNER COVERAGE BELOW

---

06 **Model and provider selection for Runlayer Assistant** NEW

55

Admins can now select the model, provider (including an OpenAI-compatible gateway), and model settings for the Runlayer Assistant from Settings → AI Providers.

— Gives exact UI path but no setting names or defaults. `1.28.1158 (2026-10-10)`

07 **Subagent context inheritance on Pi harness** IMPROVED

50

Subagents running on the Pi harness now inherit the parent agent's sandbox guidance, memory context, and skills, and can browse, use memory, and recover connectors.

— Describes capability but no config or command to invoke it. 1.28.1158 (2026-10-10)

08 **Audit events for prompt discovery** IMPROVED

45

Prompt discovery (listing a connector's prompts via Runlayer) now records audit events visible in the audit log and SIEM exports, consistent with existing tool and resource listing events.

— States outcome but no endpoint or log field detail. 1.28.1158 (2026-10-10)

09 **Upstream-sourced OAuth scope conditions** IMPROVED

45

OAuth Scopes policy conditions now always reflect scopes granted by the upstream provider, not those requested by a reconnecting MCP client.

— Describes fixed behavior with no named policy field or example. 1.28.1158 (2026-10-10)

└──▷ **BREAKING ON UPGRADE**

! ECS deployments must supply Runlayer's trusted public signing keys via the Terraform `debug_ops_request_signing_jwks` input before upgrading; the backend must be upgraded before the AI Watch endpoint or unrecognized add-ins and extensions will stay hidden.

! Kubernetes deployments must set `DEBUG_OPS_REQUEST_SIGNING_JWKS` in the `runlayer-app` Secret before upgrading; missing or unusable keys disable all diagnostic support access.

WAS THIS USEFUL? 👍 👎

OTHER / UNCATEGORIZED

◆ UNCATEGORIZED

vigolium

Feedback ⓘ v0.5.3 NOTES ↗ RANK

Vigolium - High-fidelity vulnerability scanner fusing agentic AI with native speed, modularity, and precision

> Vigolium v0.5.3 is a dense release: it adds hard budget limits and auth/preflight/triage controls to agentic scans, a finding verify/mark lifecycle, CI-friendly report formats, and OAuth2 session and multi-tab crawling support, while overhauling the CLI's JSON output into a versioned `-j` envelope and tightening scan validation, extension loading, and default server binding behind a wall of breaking changes.

└──▷ **WHAT SHIPPED · 19 FEATURES** most completely described first ? what's the number?

01 **Agent run budget limits** NEW 85

Adds `--max-commands`, `--token-budget`, and `--max-cost-usd` flags to cap agent run budgets, exiting with code 5 when a budget is exhausted.

Cap an agent autopilot run by cost and commands so runaway scans don't blow a budget.

```
$ vigolium agent autopilot --auth-file creds.yaml --max-commands
50 --token-budget 100000 --max-cost-usd 2.00 https://example.com
```

— Named flags, exit code, and a runnable example given. v0.5.3

02 **JSON output envelope overhaul (schema v2)** BREAKING 85

`schema_version` is now 2; scan commands print one scan document under `-j` instead of the previous structure, and `-j` no longer implies `--format jsonl`. A `-j` document's `command` field is always the cobra path, `import` and `js` now use the global `--format` instead of their own, and remaining read commands print the `-j` envelope (use `--json-legacy-keys` to restore old keys). `-j` output is compact by default (use `--pretty` to indent), color is automatic (`auto`) instead of always on, `replay` and `fuzz` print a human view by default with `-j` now one envelope, and `traffic -j` with tree/markdown/raw/burp output exits 2.

— Extensive named flags/keys but requires migration reading. v0.5.3

03 **Stricter scan validation and error codes** BREAKING 75

`scan <uuid>` now exits 2 as a mistyped subcommand, `-S` on `server` and `ingest` is now a usage error, `scan` / `run` with no target require `--all-records` to rescan the database, and `scan-url` / `scan-request` enforce scope without `-S` (the REST equivalent returns 422). A scan whose every target is out of scope exits 1 with `no_targets_in_scope`, a scan with no reachable targets exits 1 with `no_targets_reachable`, an unknown X-Project-UUID header now returns 404 instead of creating a project, and `doctor` exits 1 when `not_ready`.

— Many exact exit codes and error identifiers named. v0.5.3

---

04 **CI report formats: CSV, JUnit, GitHub** NEW 70

Adds `--format csv`, `junit`, and `github` report formats for CI pipelines, including emitting GitHub pull-request annotations.

Emit machine-readable CI output in GitHub annotation format to surface findings inline on a pull request.

```
$ vigolium scan https://staging.example.com --format github
```

— Named formats with a runnable example command. v0.5.3

---

05 **Extension system hardening** BREAKING 70

An `--ext` extension that fails to load now stops the scan (use `--ext-allow-failures` to restore the old behavior), `ext lint` errors on missing handlers and exports, and extension header objects are now case-insensitive, exposing `get` / `getAll` / `has` / `entries` / `raw` instead of the previous structure.

— Named flags and methods, no migration example. v0.5.3

---

06 **Agent authentication, preflight, and triage controls** NEW 65

Adds `--auth-file` to `agent swarm` and `agent autopilot` for supplying credentials to agentic runs, `--preflight-timeout` and `--no-preflight` to control agent provider ping behaviour, and `agent triage --no-write` / `--no-probe` flags plus a new `inconclusive` verdict.

— Flags and new verdict named but no example or deeper mechanism. v0.5.3

---

07 **Finding lifecycle commands** NEW 65

New `finding verify` and `finding mark --verdict` commands let users confirm whether a finding is still exploitable and record a verdict (e.g. `confirmed`) against a finding UUID.

Verify a finding is still exploitable and record the verdict before closing a ticket.

```
$ vigolium finding verify <finding-uuid> && vigolium finding mark <finding-uuid> --verdict confirmed
```

— Commands named with a runnable chained example. v0.5.3

08 **Fuzz command enhancements** NEW 60

`fuzz -r` / `--rate-limit` caps the global request rate, and `fuzz --fuzz new-param` adds a new fuzzing mode together with a `--suffix` flag and a builtin `params` wordlist.

— Several named flags and a wordlist, no usage example. v0.5.3

09 **Project and auth command changes** BREAKING 60

`project use` no longer auto-creates a project (pass `--create` explicitly), `auth load` now exits 1 on a failed login and runs the full scan login flow, and `auth totp` prints the bare code with JSON output now behind `-j`.

— Named commands and flags, behaviour changes stated plainly. v0.5.3

10 **Server binds to localhost by default** BREAKING 60

`server` now binds to `127.0.0.1` by default; exposing a non-loopback ingest proxy requires a token or allow-list.

— Clear security default change, no exact config key given. v0.5.3

THINNER COVERAGE BELOW

11 **Scan dry-run planning** NEW 55

`scan --dry-run` and `run <phase> --dry-run` print the resolved scan plan without sending any traffic.

— Exact flags named, no example run shown. v0.5.3

12 **Dismissed findings excluded from fail conditions** BREAKING 55

`--fail-on`, `scan diff`, and `--new-since` now ignore dismissed findings when evaluating results.

— Named flags/commands, no example of new behaviour. v0.5.3

13 **In-process self-update with signature verification** IMPROVED 55

`vigolium update` now updates in-process with signature verification, and `VIGOLIUM_AUTO_UPDATE` only acts when run at an interactive terminal.

— Named command and env var, no run example. v0.5.3

14 **CSV export formula injection protection** BREAKING 55

`db export --format csv` now uses standard quoting and neutralizes formula injection.

— Named flag and security behaviour, no example. v0.5.3

15 **Known-issue scan templates** NEW 45

Adds `--known-issue-scan-templates` to run specific nuclei templates.

— Single flag named, no further mechanism given. v0.5.3

16 **Scan result field semantics change** BREAKING 45

`modules_run` now counts modules that actually executed, while `modules_loaded` retains the previous count's meaning.

— Field rename explained but no usage context. v0.5.3

17 **Discovery scan tuning flags** NEW 40

Adds `--discovery-extensions` and `--discovery-max-depth` flags to tune discovery scans.

— Flags named with no behavioural detail. v0.5.3

18 **OAuth2 session support** NEW 10

The release summary states OAuth2 sessions were added, but no specific flags, config keys, or endpoints are documented in this window.

— Mentioned only in summary; no concrete surface given. v0.5.3

19 **Multi-tab crawling** NEW 10

The release summary states multi-tab crawling was added, but no specific flags or configuration are documented in this window.

— Mentioned only in summary; no concrete surface given. v0.5.3

└──▷ **BREAKING ON UPGRADE**

! `schema_version` is now 2; scan commands print one scan document under `-j` instead of the previous structure.

! `-j` on a scan no longer implies `--format jsonl`.

! A `-j` document's `command` field is always the cobra path.

- ! `import` and `js` now use the global `--format` instead of their own.
- ! Remaining read commands print the `-j` envelope; use `--json-legacy-keys` to restore old keys.
- ! `-j` output is compact by default; use `--pretty` to indent.
- ! Color is automatic (`auto`) instead of always on.
- ! `replay` and `fuzz` print a human view by default; `-j` is now one envelope.
- ! `traffic -j` with tree/markdown/raw/burp output exits 2.
- ! `agent audit` now requires `--source` .
- ! An `--ext` extension that fails to load stops the scan; use `--ext-allow-failures` to restore the old behavior.
- ! `ext lint` errors on missing handlers and exports.
- ! Extension header objects are now case-insensitive, exposing `get` / `getAll` / `has` / `entries` / `raw` instead of the previous structure.
- ! `project use` no longer auto-creates a project; pass `--create` explicitly.
- ! `auth load` exits 1 on a failed login and runs the full scan login flow.
- ! `auth totp` prints the bare code; JSON output is now behind `-j` .
- ! `--fail-on` , `scan diff` , and `--new-since` now ignore dismissed findings.
- ! `scan <uuid>` exits 2 as a mistyped subcommand.
- ! `server` binds to `127.0.0.1` by default; a non-loopback ingest proxy now requires a token or allow-list.
- ! `-S` on `server` and `ingest` is now a usage error.
- ! `scan` / `run` with no target require `--all-records` to rescan the database.
- ! `scan-url` / `scan-request` enforce scope without `-S` ; the REST equivalent returns 422.
- ! A scan whose every target is out of scope exits 1 with `no_targets_in_scope` .
- ! A scan with no reachable targets exits 1 with `no_targets_reachable` .
- ! `modules_run` now counts modules that executed; `modules_loaded` holds the old number.
- ! `vigolium update` now updates in-process with signature verification.
- ! `VIGOLIUM_AUTO_UPDATE` only acts at an interactive terminal.
- ! `db export --format csv` now uses standard quoting and neutralizes formula injection.
- ! An unknown X-Project-UUID header now returns 404 instead of creating a project.
- ! `doctor` exits 1 when `not_ready` .

WAS THIS USEFUL?  

// END OF TRANSMISSION

▷ Subscribe Feedback on this issue