CIRCL
Computer Incident
Response Center
Luxembourg

GCVE.eu

Next Steps and Future of GCVE

Ongoing work, not a prospective roadmap

https://gcve.eu/ – https://vulnerability-lookup.org

Alexandre Dulaunoy – info@gcve.eu - @→ @gcve@social.circl.lu

GCVE Workshop Vulnopticon 2026

CIRCL https://www.circl.lu

TLP:CLEAR

The Future Is Already Running

This is not a speculative roadmap

The next phase of GCVE is already being

implemented, tested and discussed.

The question is less “what could GCVE become?” and more

which operational models should we stabilise next?

The GCVE model is expanding from identifier allocation into contribution, enrichment, disclosure-state and trust-domain workflows.

Eight active workstreams

1. **Community-proposed updates** to existing CVE records through BCP-11 / GNA 65530.
2. **CVE → MITRE ATT&CK** mapping as operational vulnerability intelligence.
3. **Richer metadata and label semantics** for more accurate ATT&CK mappings.
4. **Multilingual VLAI**: English, Russian and Chinese are already supported; French and Dutch are being added.
5. **Local Exploit Hazard**: turning global exploit signals into local, control-aware exposure estimates.
6. **FlashVuln**: researcher-first publication before vendor confirmation, with explicit state/provenance.
7. **Private GNA ecosystems** for vulnerability identification that may never become public.
8. **Embargo-aware BCP extensions** for coordinated and CRA-like disclosure processes.

These workstreams share one objective: make vulnerability information usable earlier, richer over time, and explicit about its provenance and disclosure state.

The direction: from a record to a living evidence graph

Identifier

⇓

Initial advisory / vulnerability record

⇓

Community contributions + machine enrichment + operational observations

⇓

Public / embargoed / private views under explicit policy

- The vulnerability does not stop evolving when an identifier is allocated.
- New evidence should be additive, attributable and reviewable.
- Different consumers may resolve the same evidence differently without destroying provenance.

BCP-11: Community Contributions to CVE Records

BCP-11: a community update layer for existing CVEs

- BCP-11 defines a mechanism for community-proposed corrections, enrichments and extensions to existing CVE records.[1]
- Reserved **GNA 65530** is dedicated to this purpose.
- The model explicitly **does not replace the authoritative CVE record** or the responsible CNA.
- Proposals can cover affected products, version ranges, CPE/PURL, descriptions, references, severity, classifications or GCVE extensions.

The key idea is simple: useful technical corrections should not disappear merely because the original record cannot be updated quickly.

[1] https://discourse.ossbase.org/t/gcve-bcp-11-community-proposed-updates-to-existing-cve-records/1110

BCP-11 evolved toward contribution fragments

- One deterministic umbrella identifier links contributions to a target CVE:

  `CVE-2026-12345` ↔ `GCVE-65530-2026-12345`

- Each proposal is a separately identifiable **contribution fragment**.
- Technical content is immutable: corrections create a new contribution that can supersede an earlier one.
- Conflicting contributions may coexist instead of being silently collapsed into a single truth.
- A fully merged "community view" is optional and policy-driven.

> Append evidence first; resolve it later. This keeps the publication layer small, auditable and decentralised.

BCP-11 contribution lifecycle

Contributor

⇓

Contribution fragment

assertion + provenance + evidence + target CVE

⇓

Review

pending / reviewed / rejected / disputed

⇓

Publication through GNA 65530

⇓

Optional CNA upstreaming / optional resolved community view

Contribution, review and resolution are deliberately separate activities.

TLP:CLEAR

The first prototype: make contribution cheap

- The ongoing prototype focuses on a deliberately small contribution workflow.
- A contributor proposes a new JSON fragment for an existing CVE.
- Reviewers can accept or reject the proposed contribution.
- Accepted fragments become independently retrievable and can later feed resolved views or CNA update workflows.
- The prototype lets us test the difficult parts early: identifiers, fragment granularity, provenance, review state and conflict handling.

The experiment is not “Wikipedia for CVEs”. It is a structured, evidence-backed contribution channel with explicit authority boundaries.

From Vulnerability Description to Attack Semantics

Mapping CVEs to MITRE ATT&CK techniques

- The published work **Mapping CVEs to MITRE ATT&CK Techniques** builds a reproducible multi-label classifier from vulnerability descriptions.[2]
- The gold set contains **1,207 expert-curated CVEs** mapped to ATT&CK Enterprise techniques.
- The work showed that adding LLM-generated labels with low expert agreement did **not** reliably improve the classifier.
- The practical conclusion is important for GCVE: **label quality and provenance matter more than simply increasing volume**.

> Machine enrichment is useful when we can explain the training source, the label semantics and the confidence of the resulting assertion.

[2] https://arxiv.org/abs/2607.25572

From paper to Vulnerability-Lookup prototype

CVE / GCVE description → local classifier → ranked ATT&CK techniques → Vulnerability-Lookup view/API

- The research is already prototyped in Vulnerability-Lookup.
- ATT&CK mappings add attacker-behaviour context to vulnerability records without changing their identity.
- The mapping can remain an attributed assertion rather than being presented as universal ground truth.
- This creates a bridge between vulnerability handling and CTI workflows.

Repository: `https://github.com/vulnerability-lookup/cve-attack-mapping-paper`

Why the next research step is “beyond the description”

- A vulnerability description alone does not contain every signal useful for ATT&CK mapping.
- The follow-up work, **Beyond the Description: Structured Metadata and Label Semantics for CVE-to-ATT&CK Mapping**, is an active working draft.[3]
- Instead of adding more weakly labelled examples, it adds **more information per example and per label**.
- Current experiment plan includes CVSS components, CWE, affected-product CPE, inference-time priors, and ATT&CK technique-description semantics.

> The next improvement is semantic: combine structured vulnerability metadata with the meaning of the ATT&CK labels themselves.

[3] https://github.com/vulnerability-lookup/cve-attack-metadata-paper

A broader enrichment model for GCVE

Vulnerability facts

- Description
- Product / versions
- CVSS
- CWE
- CPE / PURL

Derived intelligence

- ATT&CK techniques
- Severity prediction
- Exploit signals
- Relationships
- Prioritisation

Trust metadata

- Source
- Model
- Evidence
- Confidence
- Review state

> The useful object is not just an enriched record; it is an enriched record whose derivation remains inspectable.

Local Exploit Hazard

Local Exploit Hazard: from global signal to local exposure

- Vulnerability-Lookup now includes an experimental implementation of the **Local Exploit Hazard** model.[4]
- The model starts from an exploit-likelihood input such as EPSS and adjusts it with local security controls.
- It converts the resulting probability into a **hazard rate**, allowing different time horizons and aggregate exposure views.
- The current implementation exposes the model through API primitives and notification workflows, including batch evaluation and standing exposure.

This moves prioritisation from "how severe is this vulnerability globally?" toward "what does this vulnerability contribute to my current exposure?"

[4] https://www.vulnerability-lookup.org/2026/08/11/local-exploit-hazard/

Local Exploit Hazard: open research questions

- The implementation is useful today, but should still be treated as an **open research experiment**, not as a calibrated incident predictor.
- The most important missing link is the organisation-specific layer: asset instances, exposure, control effectiveness and telemetry.
- Current follow-up questions include Weibull aggregation, whether age is counted twice when EPSS already captures temporal signals, and how to model correlated vulnerabilities or shared controls.
- The next operational step is remediation simulation: estimating which patch, upgrade or mitigation removes the most hazard per unit of effort.

GCVE can carry the identifiers and evidence; Vulnerability-Lookup can test how those records behave when transformed into local exposure models.

Multilingual Vulnerability Intelligence

VLAI next step: extend multilingual coverage

- The current VLAI work demonstrates that specialised models can support vulnerability classification efficiently.
- VLAI is already supporting **English**, **Russian** and **Chinese**.
- The next active step is to add **French** and **Dutch** models to cover additional national and regional advisory sources.
- The goal is not merely translation: classification should work directly on the language in which the technical advisory was written.

> A multilingual vulnerability ecosystem should not require English normalisation before useful machine assistance becomes possible.

Why multilingual models belong in the GCVE model

- GNAs are autonomous and may naturally publish in their local language.
- Local-language processing preserves technical nuance and reduces an unnecessary translation dependency.
- Assertions can still be shared with model/language provenance attached.
- Multiple language-specific predictions can coexist and be compared instead of forcing a single canonical machine output.

EN / RU / ZH / FR / NL advisory → language-specific VLAI → attributed enrichment

New Disclosure and Publication Models

FlashVuln: researcher-first publication

- **FlashVuln** explores fast publication by a security researcher before vendor confirmation is available.
- The publication must make the state explicit: **researcher assertion, not vendor-confirmed fact**.
- Evidence, reproducibility, timestamps, contact/coordination attempts and later vendor responses can be appended over time.
- Confirmation, dispute, remediation and identifier relationships become lifecycle updates instead of reasons to delay the first attributable record indefinitely.

Speed and trust are not opposites if the record clearly distinguishes who is asserting what, based on which evidence, at which stage of coordination.

FlashVuln: model the uncertainty instead of hiding it

Research finding

⇓

Evidence-backed researcher publication

unconfirmed / coordination initiated

⇓

Vendor / maintainer response

confirmed / disputed / needs clarification

⇓

Remediation + final public state

The important design problem is not how to pretend uncertainty does not exist; it is how to represent it safely and machine-readably.

Embargo-aware vulnerability records

- We are also working on a BCP extension for **embargo-related vulnerability handling**.
- The objective is to represent coordinated disclosure state without forcing premature public disclosure.
- The model must support CRA-like workflows where information moves through reporting, coordination, remediation and eventual disclosure stages.
- Embargo metadata should describe policy and lifecycle while keeping the sensitive technical payload within the authorised trust domain.

An identifier can exist before public disclosure; publication state and visibility are separate properties from vulnerability identity.

Candidate embargo lifecycle

Private report $\rightarrow$ **Identifier allocated** $\rightarrow$ **Embargoed coordination** $\rightarrow$ **Remediation / readiness**

$\Downarrow$

Public disclosure *or* **continued restricted handling**

- Candidate metadata: embargo state, disclosure policy, release condition/date, provenance and coordination status.
- The technical details do not need to be globally visible for the identifier and lifecycle to remain internally consistent.

Private GNA Ecosystems

Private vulnerability identification is a real requirement

- Some organisations need stable vulnerability identifiers for systems or products that cannot be publicly documented.
- Example environments include military, defence, sensitive industrial or other restricted operational ecosystems.
- In some cases, the vulnerability may remain private indefinitely rather than transitioning to public disclosure.
- Existing internal ticket numbers solve local tracking but do not provide the same interoperable vulnerability identity model.

> GCVE identifiers and GNA autonomy can also be useful inside a bounded trust domain.

Private GNA strategy

Keep the GCVE semantics

- Stable identifiers
- GNA-local allocation
- Provenance
- Relationships
- Structured records
- Revision history

Change the distribution model

- Restricted discovery
- Access-controlled feeds
- Organisation-defined trust domain
- No mandatory public publication
- Optional later release

The identifier model should not force a disclosure policy. Public, embargoed and permanently private vulnerability ecosystems can share the same underlying semantics.

Public and private are not two separate worlds

	Public ecosystem	Private ecosystem
Identifier	GCVE / CVE relationships	GCVE within trust domain
Publication	Open GNA feed / dump / API	Restricted feed / API
Evidence	Publicly verifiable where possible	Controlled-access evidence
Enrichment	Community + machine assertions	Internal + machine assertions
Transition	Already public	May remain private or later disclose

Interoperability comes from shared semantics, not from requiring every participant to expose the same data publicly.

Converging the Workstreams

One vulnerability, many attributable views

Base vulnerability identity

⇓

Community: BCP-11 corrections

Machine: ATT&CK / VLAI enrichment

Disclosure: researcher / embargo / private state

⇓

Consumer-specific resolution under explicit trust policy

GCVE does not need one global winner for every field. It needs enough provenance for each consumer to make an explicit resolution decision.

The architecture we are converging toward

- **Allocation**: autonomous GNAs create stable vulnerability identifiers.
- **Publication**: public, embargoed or private distribution follows the operating context.
- **Contribution**: BCP-11 lets external communities append evidence-backed proposals.
- **Enrichment**: specialised models and research outputs add severity, ATT&CK and other derived intelligence.
- **Provenance**: every assertion retains who/what produced it, from which evidence and with what review state.
- **Resolution**: tools such as Vulnerability-Lookup can construct useful views without destroying the original assertions.

The future GCVE stack is less a central database than a protocol for preserving identity, evidence and trust across vulnerability workflows.

What we are testing next

1. Make the **BCP-11 contribution prototype** usable on real CVE correction cases.
2. Move CVE-to-ATT&CK research from description-only classification toward **structured metadata and label semantics**.
3. Extend **VLAI** from English, Russian and Chinese toward French and Dutch operational workflows.
4. Continue **Local Exploit Hazard** research: calibration, asset-instance semantics, control-effectiveness evidence and remediation simulation.
5. Define the state/provenance model for **FlashVuln researcher-first publication**.
6. Formalise **embargo metadata** for coordinated and CRA-like disclosure processes.
7. Validate a **private GNA deployment model** for restricted vulnerability ecosystems.

The next BCPs should emerge from these implementations and operational constraints, not the other way around.

Takeaways

1. **GCVE is moving beyond allocation**: contributions, enrichment and disclosure state are becoming first-class concerns.
2. **BCP-11 makes corrections additive**: immutable contribution fragments preserve provenance and disagreement.
3. **Vulnerability intelligence is getting richer**: ATT&CK mapping, multilingual VLAI and local exploit hazard add operational context while remaining attributable.
4. **Disclosure is not binary**: researcher-first, embargoed and private ecosystems need explicit lifecycle models.
5. **The common denominator is trustable provenance**: who asserted what, when, based on which evidence, and under which visibility policy.

Thank you for your attention

- GCVE: https://gcve.eu/
- BCP-11 discussion: https://discourse.ossbase.org/t/gcve-bcp-11-community-proposed-updates-to-existing-cve-records/1110
- CVE → ATT&CK paper: https://arxiv.org/abs/2607.25572
- ATT&CK mapping repository: https://github.com/vulnerability-lookup/cve-attack-mapping-paper
- Follow-up research: https://github.com/vulnerability-lookup/cve-attack-metadata-paper
- Local Exploit Hazard: https://www.vulnerability-lookup.org/2026/08/11/local-exploit-hazard/
- Vulnerability-Lookup: https://vulnerability-lookup.org/
- Questions: info@gcve.eu
- GCVE **@gcve@social.circl.lu**