Introducing CVE-DIBS

Howdy, CNA partners! Have you ever seen a public vulnerability hit the news or your intelligence feeds and wondered when a CVE ID might be assigned so organizations can track and remediate risk? We have, too!

Let us introduce ourselves. We're the recently established Researcher Working Group (RWG), a flock of researcher and other CNAs who often find, analyze, and report vulnerabilities in a wide range of third-party software. Many of us have worked for both Supplier (vendor) CNAs and CNAs who assign CVEs to third-party systems and services. Research-oriented CNAs commonly see a variety of supplier responses and disclosure practices firsthand, particularly for exploited and other time-sensitive issues.

As one of our inaugural activities, the RWG has been collaborating with the rest of the CVE Program to create an experimental protocol for coordinating CVE assignment when we observe a publicly disclosed vulnerability that does not have a CVE and is in urgent need of an identifier. As it's scoped now, the protocol, which we've named "CVE DIBS," is intended to accelerate CVE assignment for "hot" vulnerabilities in the news and in the wild while still supporting CVE data quality and reducing risk of collisions.

DIBS is currently in **closed beta** and is intended to speed up CVE assignment by providing a dedicated, GitHub-based channel to surface time-sensitive public vulnerabilities and coordinate CVE publication across CNAs and TLRs. Additional detail on how to use CVE-DIBS is below!

How to use CVE-DIBS

Step 0: A CNA (CNA1) with appropriate scope spots a recent vulnerability without a CVE in a news headline, in a community discussion with a large audience (e.g., on Hacker News), or being exploited in the wild. CNA1 ascertains that the affected product(s) **are not** in another CNA's scope. CNA1 decides this vulnerability needs a CVE and quick; but first, they want to make sure a CVE isn't already in the works from another CNA (including CNAs of Last Resort, or CNA-LRs)). CNA1 can now become a CVE-DIBS **Requestor.**

1. **Requestor** (**CNA1**) posts an issue in the DIBS repo noting that they **intend** to assign a CVE to the public vulnerability and would like to know if another CNA has already been working with the supplier, vulnerability finder, or TLR (etc) to assign a CVE.
    a. CNA1 may *optionally* also surface the DIBS request in #research-dibs (Slack)
2. CNA1 waits 8 hours to ensure no DIBS collision response; if another CNA or TLR indicates a CVE is already in progress, CNA1 may request an expected CVE publication timeline from CNA2 or TLR.
    a. Other CNAs who wish to signal they have no prior claim can *optionally* respond with a "DIBS-Agree" comment on the original DIBS issue.
3. If there is no DIBS collision explicitly noted by the time 8 hours have passed, CNA1 assigns and publishes a CVE.

4. If a DIBS collision does exist, CNA1 and CNA2 (or TLR) discuss competing claims and publication timelines; the claimants decide amongst themselves who the ultimate “designee,” or **assigning CNA,** will be. The designee assigns and publishes a CVE.
5. Assigning CNA SHOULD publish a CVE record to the CVE list within 2 hours of final DIBS decision. Assigning CNA MUST publish a CVE Record to the CVE List within 24 hours of final DIBS decision.
    a. A CNA-LR MAY delegate assignment to R or another appropriate CNA participating in the Dibs-Request.

Who can participate in CVE-DIBS?

In order to be added to the experimental repository, you must be:

- A current CNA in good standing;
- Have broad CNA scope (this often equates to researcher or open-source scope, but may also include CNAs with vendor scope who find themselves working with other CNAs to assign CVEs to third-party products on a regular basis); **and**
- Be **nominated** and **seconded** by other CNAs involved in the DIBS process

Longer-term, we expect to open up CVE-DIBS to more participants and are considering whether the repository should be public for the sake of transparency and community awareness.

FAQ

How is “urgent” defined?
We’ve been back and forth on this ourselves in RWG discussions. “Urgent” or “hot” vulnerabilities should be **recent** and often have one or more of the following traits:

- Exploitation in the wild and/or public exploit code
- Coverage in security news headlines
- Community pickup on social media
- The vulnerability occurs in a product or service that has previously been targeted by adversaries in major real-world incidents

In research CNA parlance, you often know “hot” vulns when you see them. If a public vulnerability is making the rounds on Reddit or in security news headlines and has proof-of-concept code with a bunch of stars on GitHub, it’s a good bet that CISOs and analysts will be paying attention. It’s extremely helpful for these vulnerabilities to get CVEs quickly so they can be operationalized and assessed as part of VM and other toolchains.

What if there’s very little information available about the public vulnerability?
This experimental channel will let us test the boundaries of what makes a reasonable “DIBS” call and what needs more discussion before we can assign a CVE. In general, if a patch or hotfix is available (or in a commit somewhere discoverable even if it’s not generally available), we should assume adversaries can and will find it. Similarly, if a supplier emphasizes the importance of a security patch in public or customer communications, that signals CVE urgency.