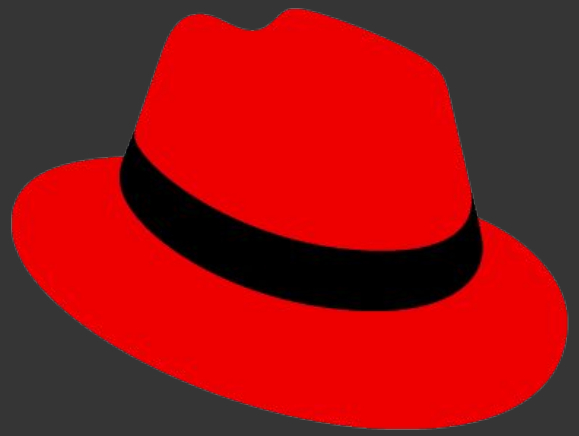
Security Vulnerability Reporting: Who to Trust?



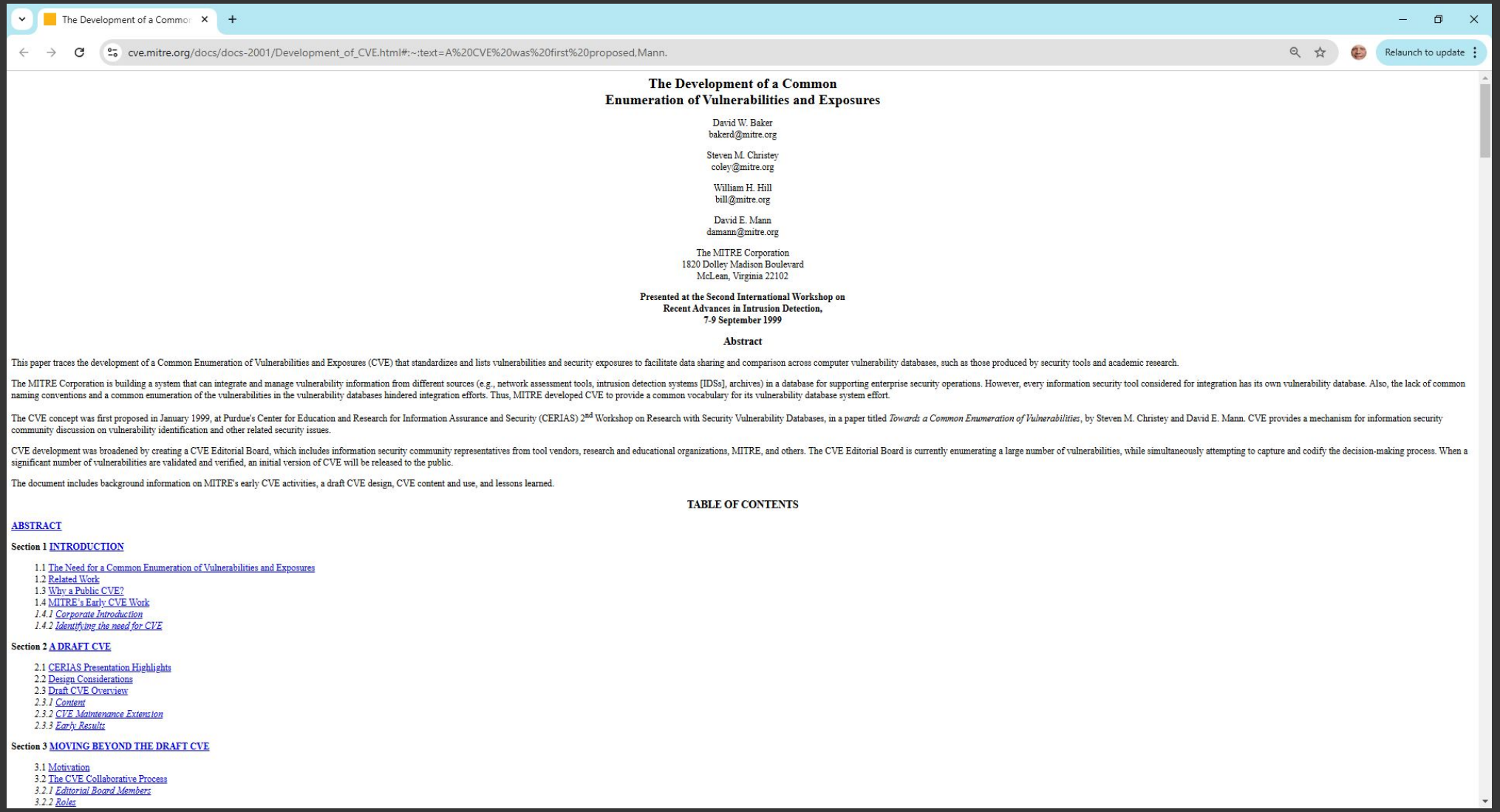
Red Hat

Greg Scott – gscott@redhat.com
CISSP Number 358671

Central Indiana LUG
January 8, 2025

Trust but verify

Vulnerability reporting has come a long way since 1999.



The Development of a Common

cve.mitre.org/docs/docs-2001/Development_of_CVE.html#:~:text=A%20CVE%20was%20first%20proposed,Mann.

Relaunch to update

The Development of a Common Enumeration of Vulnerabilities and Exposures

David W. Baker
bakerd@mitre.org

Steven M. Christey
coley@mitre.org

William H. Hill
bill@mitre.org

David E. Mann
damann@mitre.org

The MITRE Corporation
1820 Dolley Madison Boulevard
McLean, Virginia 22102

Presented at the Second International Workshop on Recent Advances in Intrusion Detection, 7-9 September 1999

Abstract

This paper traces the development of a Common Enumeration of Vulnerabilities and Exposures (CVE) that standardizes and lists vulnerabilities and security exposures to facilitate data sharing and comparison across computer vulnerability databases, such as those produced by security tools and academic research.

The MITRE Corporation is building a system that can integrate and manage vulnerability information from different sources (e.g., network assessment tools, intrusion detection systems [IDSs], archives) in a database for supporting enterprise security operations. However, every information security tool considered for integration has its own vulnerability database. Also, the lack of common naming conventions and a common enumeration of the vulnerabilities in the vulnerability databases hindered integration efforts. Thus, MITRE developed CVE to provide a common vocabulary for its vulnerability database system effort.

The CVE concept was first proposed in January 1999, at Purdue's Center for Education and Research for Information Assurance and Security (CERIAS) 2nd Workshop on Research with Security Vulnerability Databases, in a paper titled *Towards a Common Enumeration of Vulnerabilities*, by Steven M. Christey and David E. Mann. CVE provides a mechanism for information security community discussion on vulnerability identification and other related security issues.

CVE development was broadened by creating a CVE Editorial Board, which includes information security community representatives from tool vendors, research and educational organizations, MITRE, and others. The CVE Editorial Board is currently enumerating a large number of vulnerabilities, while simultaneously attempting to capture and codify the decision-making process. When a significant number of vulnerabilities are validated and verified, an initial version of CVE will be released to the public.

The document includes background information on MITRE's early CVE activities, a draft CVE design, CVE content and use, and lessons learned.

TABLE OF CONTENTS

See **https://www.cve.org/About/History**

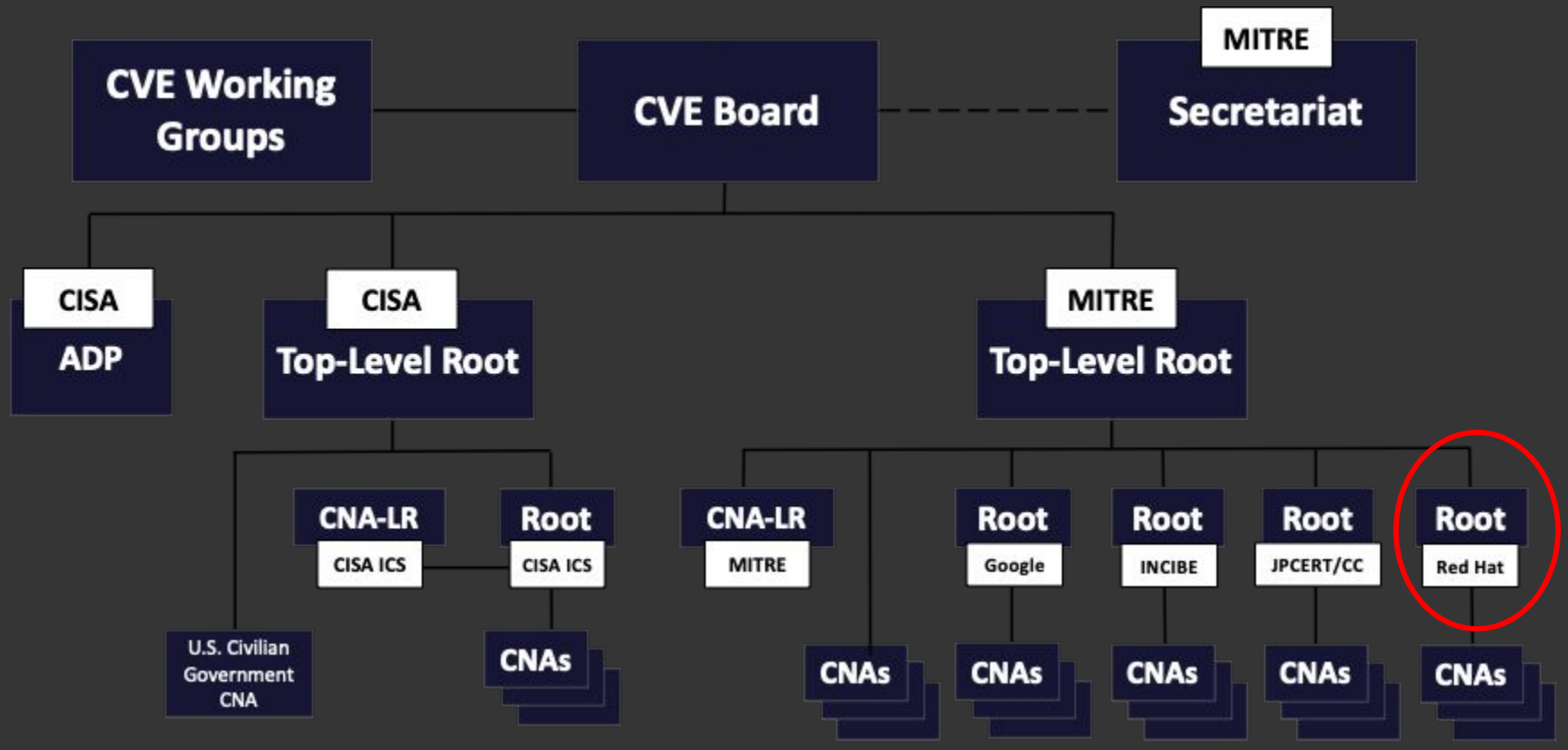
and **https://cve.mitre.org/docs/docs-2001/Development_of_CVE.html**

Today - CVE.org

- Identify, define, and catalog publicly disclosed cybersecurity vulnerabilities.

- As of Jan. 2, 2025, 240,830 CVE Records accessible via Download or Search

Today's Players



Definitions - See https://www.cve.org/ResourcesSupport/Glossary?activeTerm=glossaryRoot

- **CVE** - Common Vulnerabilities and Exposures.
- **CNA** - CVE Numbering Authority. The authoritative source for a CVE record.
- **CNA-LR** - CVE Numbering Authority of Last Resort. Resolves disputes within its governance area.
- **Root** - An organization authorized within the CVE Program that is responsible, within a specific Scope, for the recruitment, training, and governance of one or more entities that are a CNA, CNA-LR, or another Root.
- **TL-Root** - Top Level Root, responsible for governance and administration of its hierarchy, including Roots and CNAs within that hierarchy.
- **ADP** - Authorized Data Publisher. The ADP role enables a qualified and authorized organization to enrich the content of CVE Records published by a CVE Numbering Authority (CNA) with additional, related information (e.g., risk scores, references, vulnerability characteristics, translations, etc.).
  See https://www.cve.org/ProgramOrganization/ADPs.

More Definitions

- **CISA** - United States Cybersecurity and Infrastructure Security Agency.
  See https://www.cisa.gov/.

- **MITRE Corporation** - operates 6 of 42 federally funded research and development centers (FFRDCs) supporting various U.S. government agencies in the aviation, defense, healthcare, homeland security, and cybersecurity fields, among others.
  - In 1999 MITRE and top security organizations created CVE®, the first public dictionary of computer vulnerabilities to boost cyber defense.
  - Today - handles secretariat and support duties for The CVE Program.
  - See https://www.mitre.org/who-we-are/our-story

A definition biggie

Z-stream - Red Hat numbers product versions as x.y.z, kind of like the old Dewey Decimal System. X is the major version, Y is the minor version. Think of a z-stream as a patch level.

- Red Hat supports multiple concurrent product version streams.
- Think of a car company when the new model year comes out. What if they backported new model improvements into earlier model years?
- That makes no sense with cars. But Red Hat backports bug fixes from new software versions to older versions when feasible.
- ***Which means a new z-stream from an old version might have fixes you haven't yet applied in an old z-stream from a new version.***

CVE Lifecycle

1. **Discover:** somebody discovers a new vulnerability.
2. **Report:** discoverer reports a vulnerability to a CVE Program partner.
3. **Request:** the CVE Program partner assigns a CVE Identifier (CVE ID).
4. **Reserve:** automation assigns the initial state of this CVE Record as "*Reserved*."

   The *Reserved* state means that CVE stakeholder(s) are using the CVE ID for early-stage vulnerability coordination and management, but the CNA is not yet ready to publicly disclose the vulnerability.

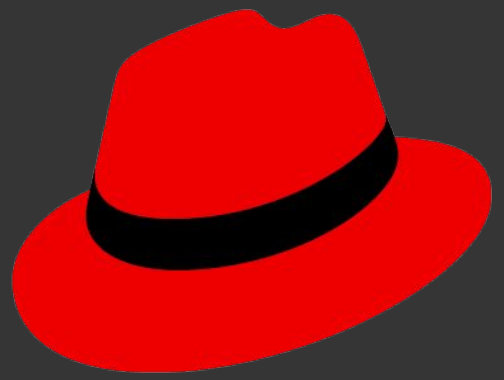
5. **Submit:** CVE Program partner submits the details.

   Details include but are not limited to affected product(s); affected or fixed product versions; vulnerability type, root cause, or impact; and at least one public reference.

6. **Publish:** once the CVE record includes the minimum required data elements, the responsible CNA publishes it to the CVE List.

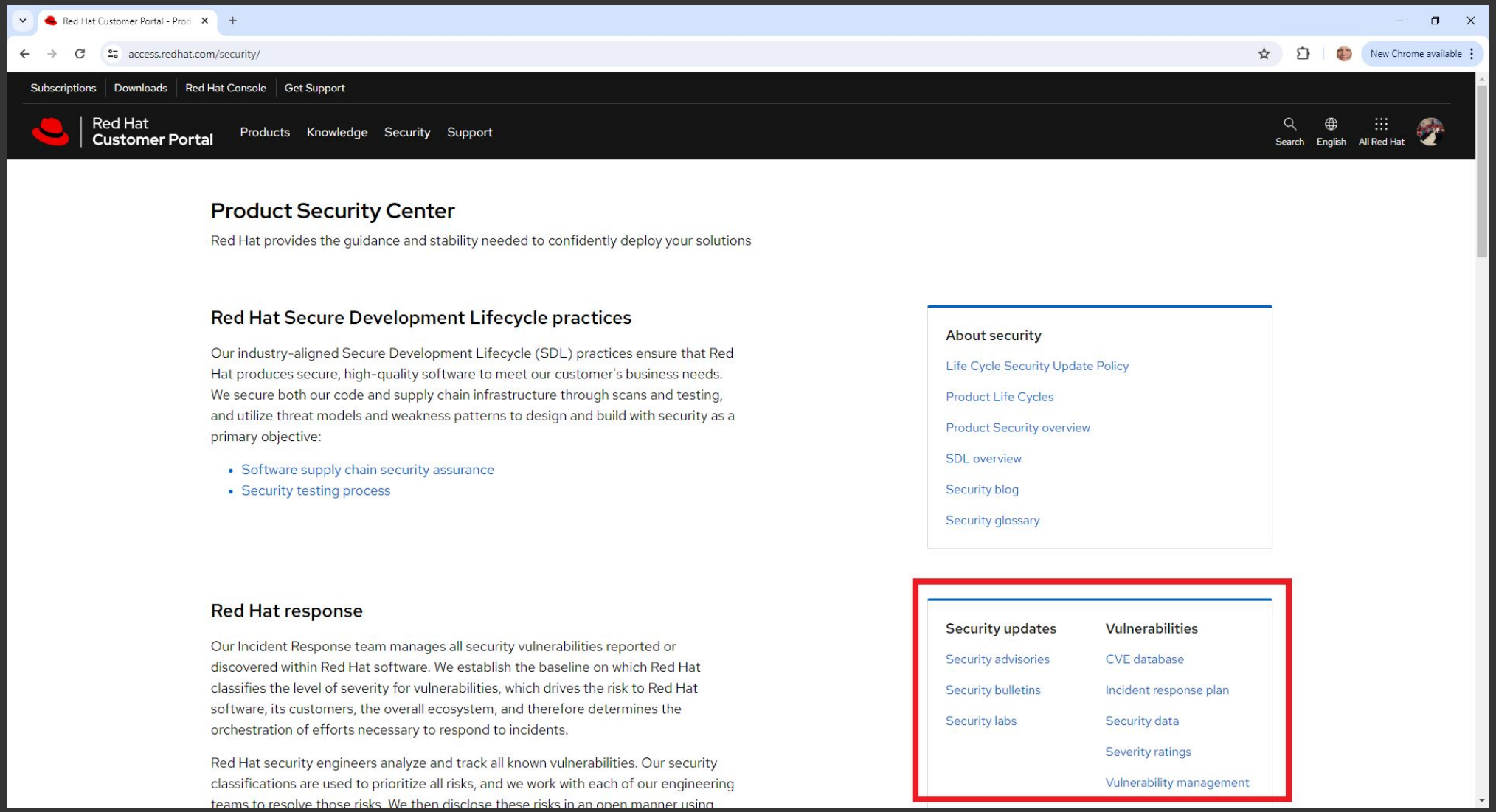
The CVE Record is now available for download and viewing by the public.

See https://www.cve.org/About/Process



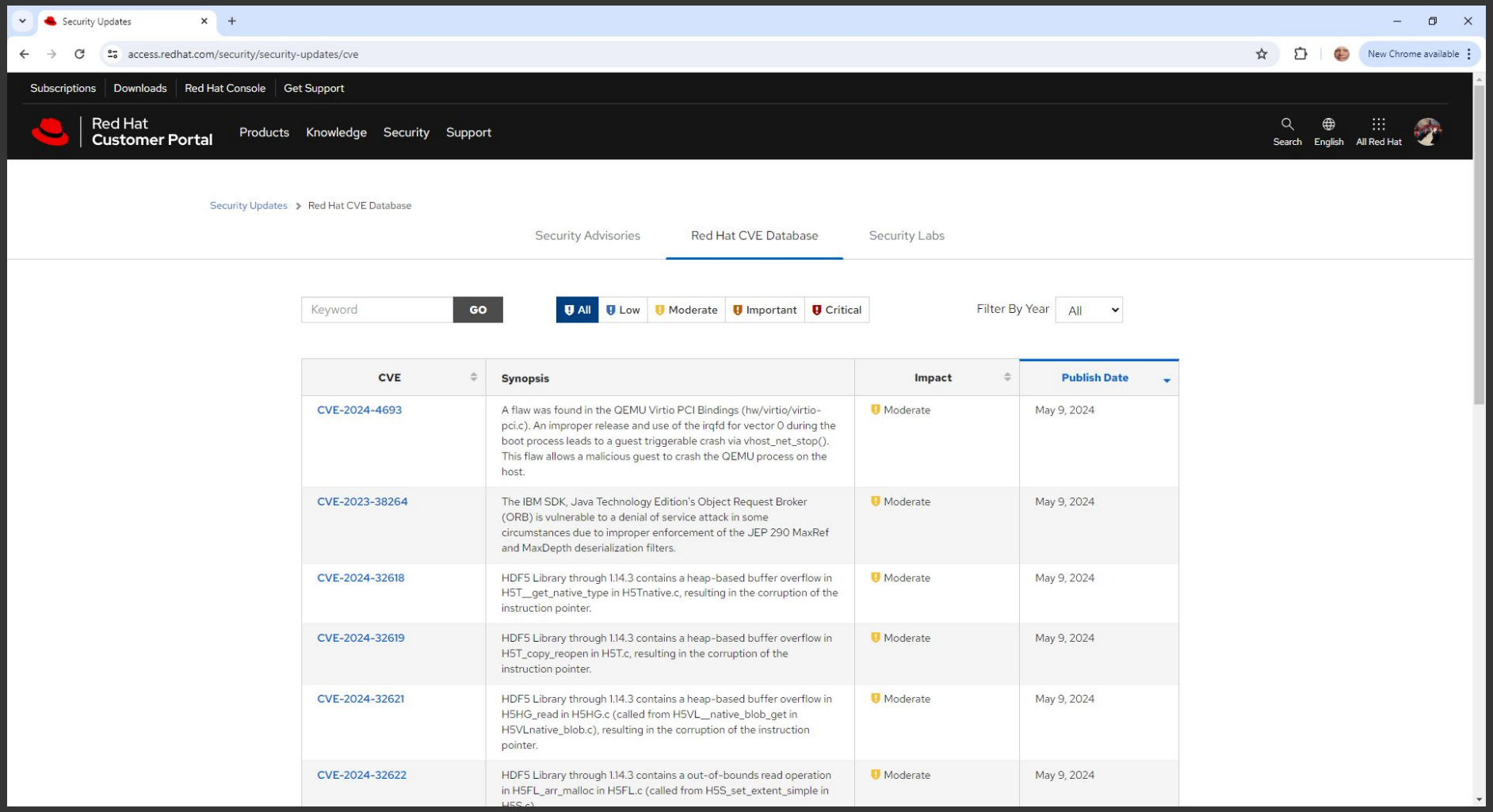
Security Repository

Red Hat Security Pages



- One URL for all the good stuff.
- https://access.redhat.com/security/

Human-readable CVE Information.



- CVE - **C**ommon **V**ulnerabilities and **E**xposures. Think security vulnerabilities.
- Red Hat triage teams deal with multiple CVEs every day.
- That's why staying current is critical.

Machine Readable CVE Data

- Enter VEX files with these possible states.
  - **Fixed**: With a link to the released CSAF-VEX advisory
  - **Known Affected**: Confirmation that the specific product and component is affected by a particular CVE
  - **Known Not Affected**: Confirmation that the specific product and component is NOT affected by a particular CVE
  - **Under Investigation**: Information that the Red Hat Product Security team is verifying the applicability (and its impact) of a specific CVE to a particular product and component

- OVAL (Open Vulnerability and Assessment Language) does not assess risk.
  - Retired end of 2024; supported through RHEL 9.

- See https://www.redhat.com/en/blog/future-red-hat-security-data
- And https://www.redhat.com/en/blog/security-vulnerability-reporting-who-can-you-trust
- And https://www.redhat.com/en/blog/red-hat-vex-files-cves-are-now-generally-available

Bug Hunting Stories

Yes, I know… Corny

Kernel bug

Side Channel attacks - Spectre/Meltdown - 2017

- Collaboration at its best - see https://meltdownattack.com/
- Three teams independently discovered and reported Meltdown:
  - Jann Horn (Google Project Zero),
  - Werner Haas, Thomas Prescher (Cyberus Technology),
  - Daniel Gruss, Moritz Lipp, Stefan Mangard, Michael Schwarz (Graz University of Technology)
- Two people independently discovered and reported Spectre:
  - Jann Horn (Google Project Zero) and Paul Kocher in collaboration with, in alphabetical order, Daniel Genkin (University of Pennsylvania and University of Maryland), Mike Hamburg (Rambus), Moritz Lipp (Graz University of Technology), and Yuval Yarom (University of Adelaide and Data61)

More Spectre/Meltdown credit

- First reported to Intel and other chip makers June 1, 2017
- That led to a mad scramble behind the scenes to address it.
- Went public Jan. 3, 2018, one week earlier than planned, after an article appeared in The Register.
- And that led to another mad scramble to get the updates out.
- See this article from Wired Magazine (Andy Greenburg, Jan. 7, 2018) for a great writeup on how researchers pieced it together: https://www.wired.com/story/meltdown-spectre-bug-collision-intel-chip-flaw-discovery/

…Which led to a bunch of CVEs

- CVE-2017-5753, Spectre variant 1, Bounds Check Bypass
- CVE-2017-5754, Meltdown – variant 3, Rogue Data Cache Load

And then in 2019…

- CVE-2018-3639. Kernel Side-Channel Attack using Speculative Store Bypass
- CVE-2018-3620 & CVE-2018-3646, L1TF - L1 Terminal Fault Attack
- CVE-2018-12130, CVE-2018-12126, CVE-2018-12127, and CVE-2019-11091, MDS - Microarchitectural Data Sampling

And they keep coming.

log4Shell - 2021

Apache Log4j 2 - popular Java library for logging application error messages.

Log4Shell - a vulnerability that allowed remote code execution (RCE).

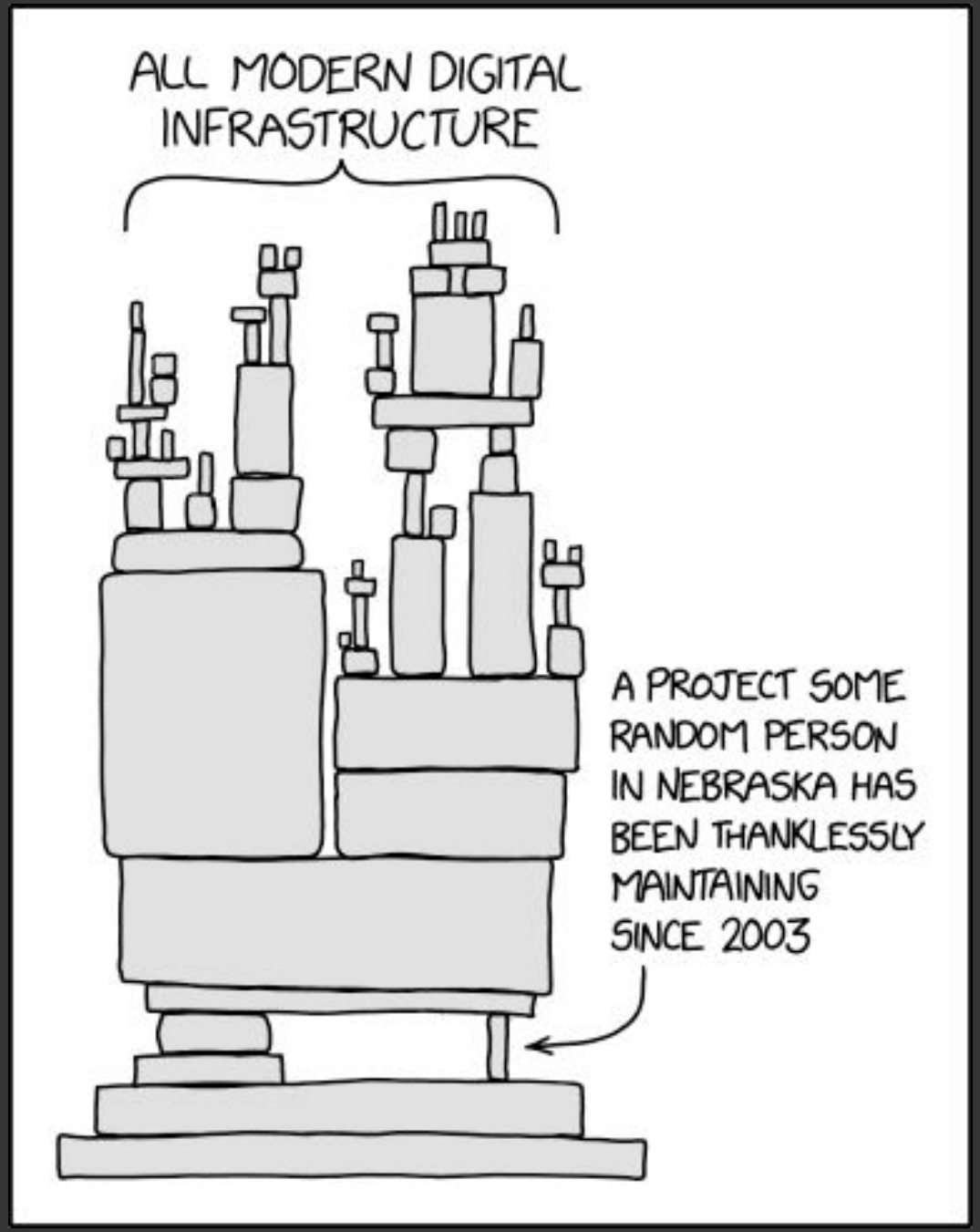
September 21, 2013 - Introduced with log4j 2.0-beta 9.

November 24, 2021 - Chen Zhaojun of Alibaba reported the vulnerability to the Apache Foundation.

November 29, 2021 - developers investigated and worked on a fix.

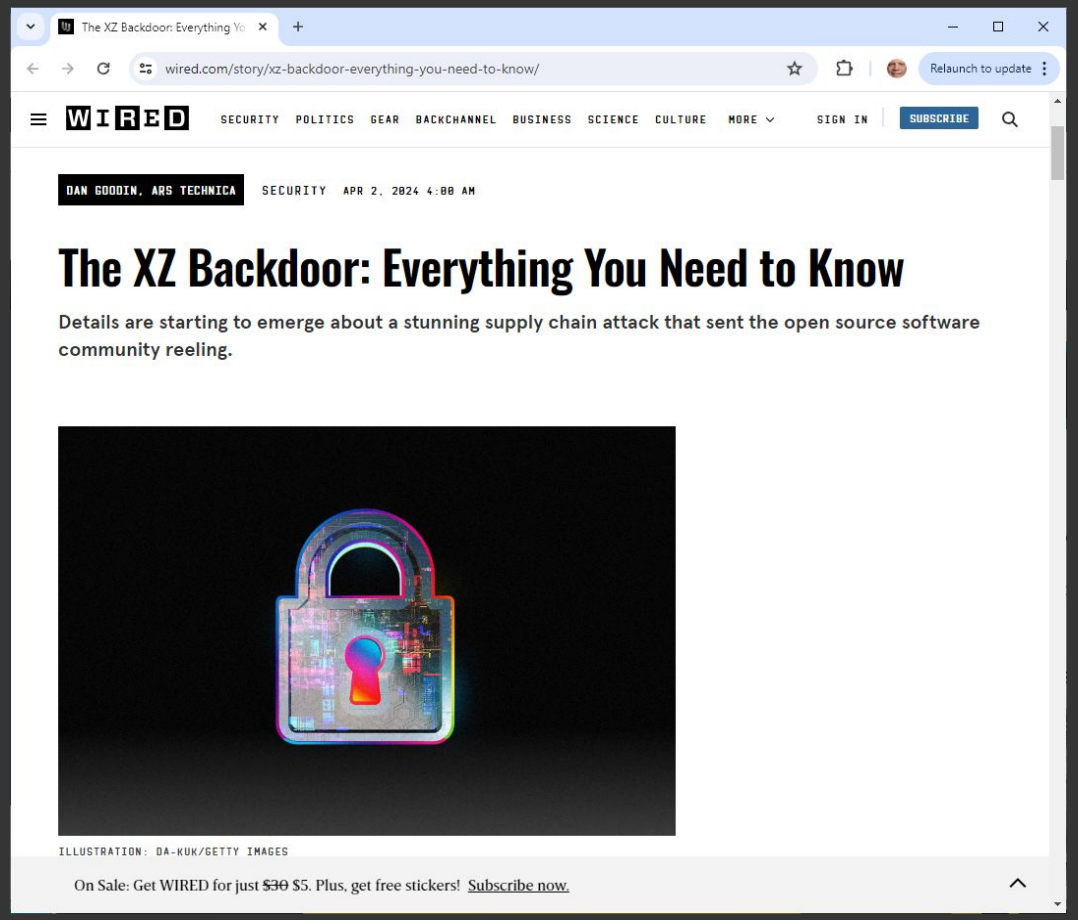
December 9, 2021 - found in the wild, public disclosure, mad scramble.

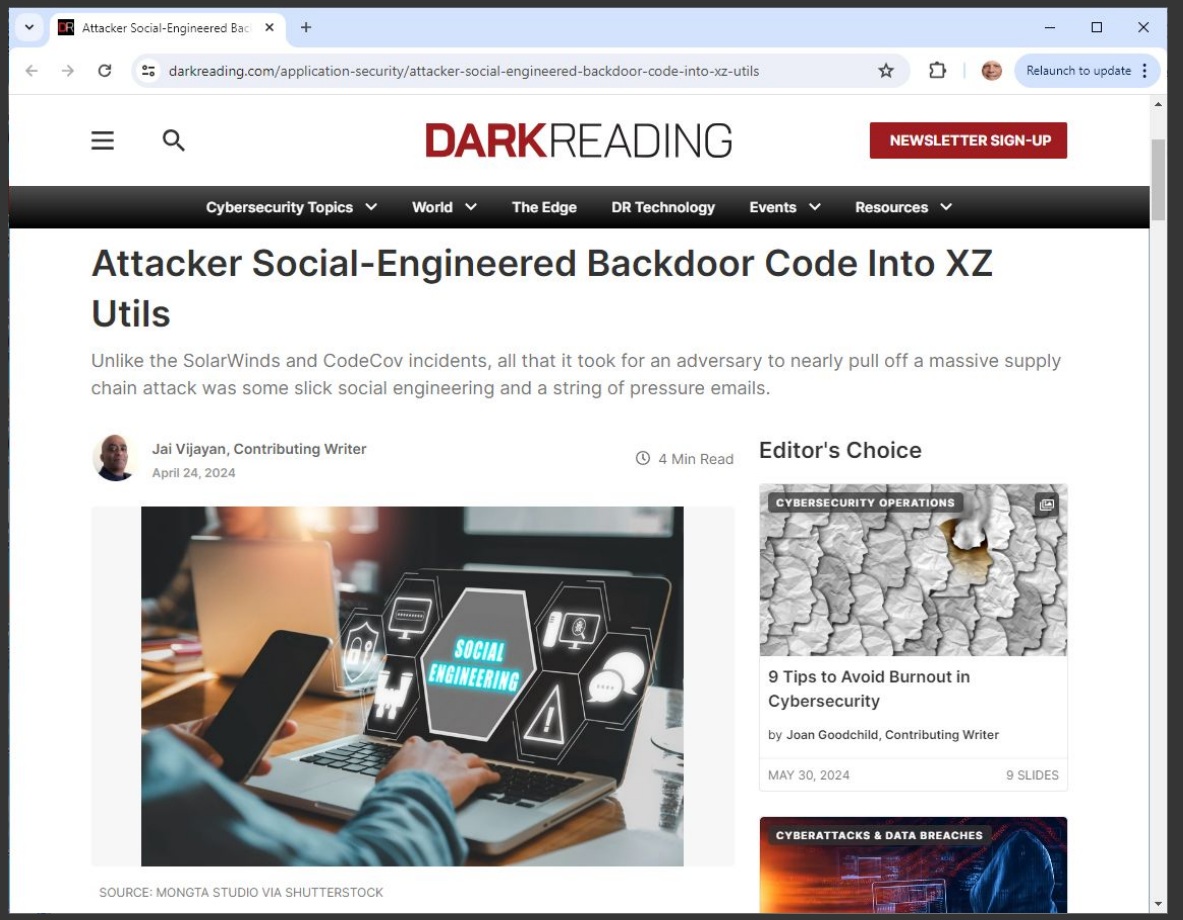
The XZ Utils attack - 2024



From https://xkcd.com/2347/

The XZ Utils attack - CVE-2024-3094





https://www.wired.com/story/xz-backdoor-everything-you-need-to-know/

https://www.darkreading.com/application-security/attacker-social-engineered-backdoor-code-into-xz-utils

XZ Utils - Sophisticated social engineering

Good guys

- Lasse Collin - longtime XZ Utils maintainer.
- Andres Freund - Microsoft PostgreSQL developer and engineer.

Bad guys

- Jia Tan - Username JiaT75 on the XZ utils email list.
- Jigar Kumar, Dennis Ens, and others.
- Nobody knows if these clowns are the same person or part of a Chinese attack group.
- Many believe they're connected.

CUPS remote code execution, Sept. 2024

See

https://www.redhat.com/en/blog/red-hat-response-openprinting-cups-vulnerabilities

Numbers

Published cve.org CVE records

2024	2023	2022	2021	2020	2019	2018	2017
40,009	28,961	25,059	20,161	18,375	17,308	16,512	14,645

CVE Record Publications by All CNAs Combined Versus CNA-LRs

Year	2024	2023	2022	2021	2020	2019	2018	2017	2016-1999
All CNAs	83%	79%	68%	65%	58%	53%	53%	50%	0%
CNA-LRs	17%	21%	32%	35%	42%	47%	47%	50%	100%

*Note: CISA became a CNA-LR in calendar year 2020.

See https://www.cve.org/about/Metrics

Red Hat CVE numbers

	2024	2023	2022	2021	2020	2019	2018	2017
CVE records	6887	1881	2075	1866	2168	2272	2223	2629
Security advisories	2935	2258	1614	1357	1552	965	823	696

See https://access.redhat.com/security/security-updates/

- Fill in date ranges, Jan. 1 through Dec. 31 of the year you want.
- Scroll to the bottom of the page for the total.

Red Hat in the big leagues

Of 432 CNA partners at the beginning of 2025,...

- Red Hat was the 17th top CNA in 2024.
- Kernel.org was number 2.
- Microsoft was 7.
- Adobe was 9.
- Apple was 10.
- IBM was 11.

Audit Scans

Typical false positive

- CVE-2023-24329, Python flaw found in a Red Hat Enterprise Linux (RHEL) 8.6 container image.
- The installed version was 3.6.8-47.el8_6.
- The scan claimed this version was broken and the fixed version was 3.6.8-51.el8_8.1.
- But the scan for this customer should have used data about the RHEL 8.6 Extended Update Service (EUS) release stream instead of the general RHEL 8 release stream.
- The RHEL 8.6 EUS errata notice showed the fixed image was platform-python-3.6.8-47.el8_6.1.i686.rpm, the same version the scan claimed was broken.
- The scan incorrectly flagged it as broken, probably because 3.6.8-47 is less than 3.6.8-51.

Library false positive

- CVE-2023-49569 - critical vulnerability with certain functions in a version of the go library, go-git.

- A recent security scan flagged all 201 product components that depend on this library, *including components that never use the vulnerable functions*.

- Especially with library vulnerabilities, distinguish false-positives from truly vulnerable components by carefully reading the relevant Red Hat security repository CVE writeups.

Severity levels

FAQs

So, who do I believe?

Q: My scan found a bunch of critical vulnerabilities. But you call them moderate or important. Why should I believe you?

A: Because Red Hat earned a trusted role in the security community.

- Red Hat triaged more than 2900 vulnerabilities in 2024. We're pretty good at it.
- See https://access.redhat.com/security for the definitive source of truth for security vulnerabilities related to Red Hat products.

Red Hat makes all this data available in both human and machine-readable form to help auditors perform the best quality scans possible.

Q: You claim you fixed a CVE in an earlier version. We scanned a later version and the scan still called it out. What gives?

A: A few possibilities.

- Z-stream backport false positive.
- The bug doesn’t apply.
- The bug is moderate and we closed with FIX DEFERRED.
  - (Which means we will probably fix it later in a Y-stream update.)
- We messed up.

Q: Why won’t you make all these bugs go away?

A:

Q: How do I become fully secure?

A: There’s no such thing as fully secure. Security is all risk management.

Fantasy picture from https://aesthetics.fandom.com/wiki/Fantasy

Security is a process, not an event.

I’m secure!

Now everything will always be all better!

From https://www.reddit.com/r/Transgender_Fotos/comments/bbfw5b/mom_and_apple_pie_apple_pie_genocide/

Use this tool to find the most important security person in your organization.

Tactics to help attackers plunder you

- Practice security by obscurity.
- Become a slave to checklists.
- Become a slave to official frameworks.
- Grow complacent.
- Toss security over the wall to an official scapegoat.
- Trust the CIA, CISA, FBI, NIST, NSA, NSF, the US Department of Homeland Security, or your favorite government agency to protect you.
- Allow tech tools to supersede old-fashioned human judgment.
- Classify your information technology as a cost instead of an asset.

If you do this…

…then sooner or later, you'll end up like this.

Thanks to…

Pete Allor, Senior Director in Red Hat Product Security, for feedback and suggestions.

Przemyslaw “Rogue” Roguski, Principal Product Security Engineer, for blog posts about VEX files and guidance around vulnerability data modeling.

A growing army of bug chasers inside Red Hat and the open source community who are really really really good with this stuff.

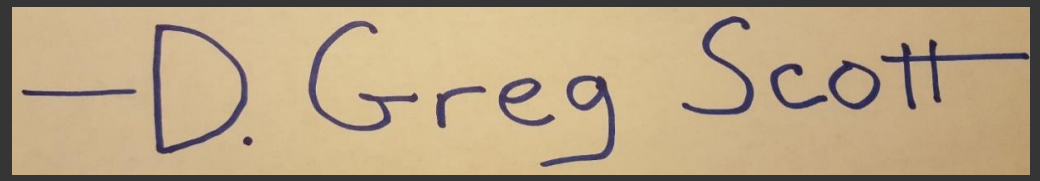
A customer base that supports it all.

Contact Info

Greg Scott – gscott@redhat.com

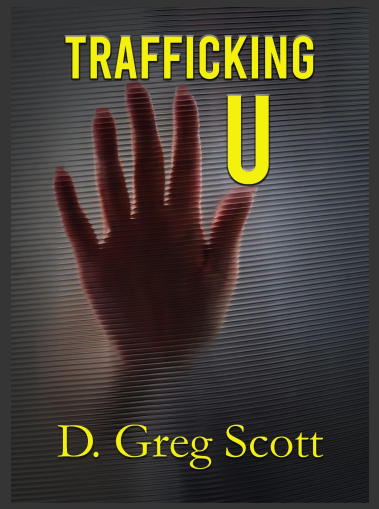
Red Hat

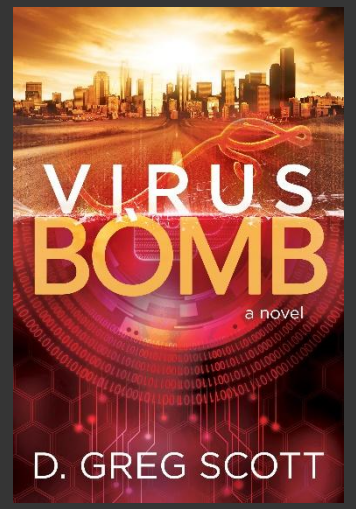
Personal contact info

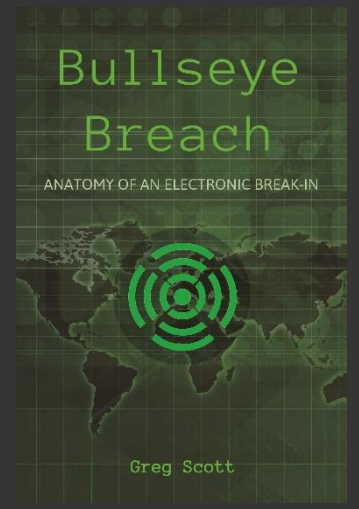


gregscott@infrasupport.com or

gregscott@dgregscott.com







https://www.dgregscott.com

Twitter: DGregScott

LinkedIn: https://www.linkedin.com/in/dgregscott/

Youtube: “D. Greg Scott Public Videos” at https://www.youtube.com/@dgregscottpublicvideos