Preparing Your OSS Project for the First Vulnerability Report

SIEMENS

Vulnerabilities, Anyone?

Who of you ever caused a vulnerability?

Did it get a CVE assigned?

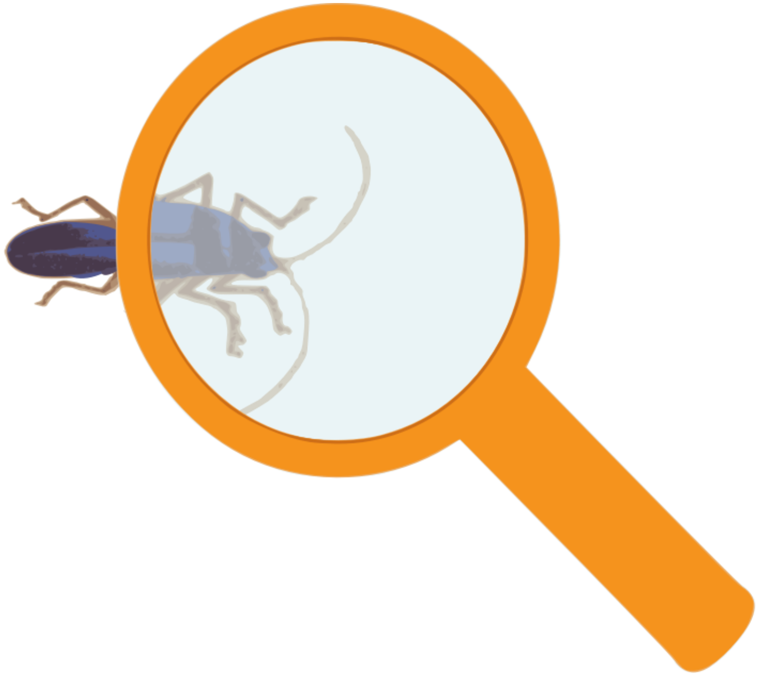
My #1: CVE-2004-0176, Multiple buffer overflows in Ethereal 0.8.13 to 0.10.2,
IRDA Dissector Plugin IRCOM_PORT_NAME Overflow (+12 others)

Who of you ever requested a CVE?

Anyone already reported a vulnerability?

How to Report a Vulnerability?

- **Coordinated vulnerability disclosure (responsible disclosure)**
  - Reach out privately first
  - Give sufficient time to validate and fix
- **Commercial product => contact vendor**
- **Open source project => ?**
  - Maintainer(s)? Via email?
  - Private issues/tickets?
  - Other channels?
  - Siemens associated projects were contacted via Product CERT

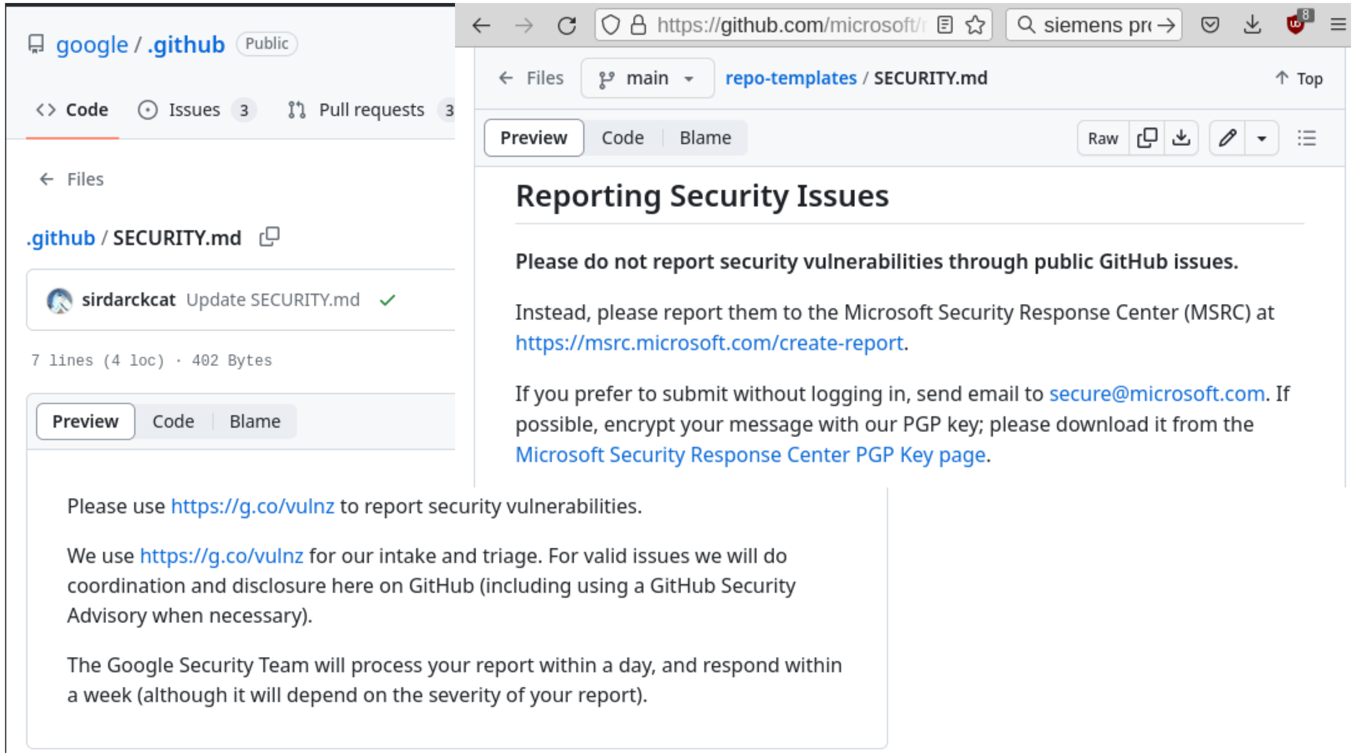


Taking Reports via Corporate Product Channels

- **Indirection may cause extra delays**
- **May involve issuing a product security advisory afterwards**
  - "*The Siemens product Jailhouse has fixed following vulnerabilities*..." - not really
- **Are "our" OSS projects products?**
- **What if it should not appear as "Siemens controlled"?**
- **What if the maintainer is leaving the company?**
- **What if the project is handed over to a third party?**

What Do Others Do?

- **Microsoft**
  - Very professional
  - But very "corporate" as well
- **SAP**
  - Seems to have copied from MS...
- **Google**
  - Leaner
  - Own infrastructure for reporting
  - Then GitHub advisories



Reporting Security Issues

Please do not report security vulnerabilities through public GitHub issues.

Instead, please report them to the Microsoft Security Response Center (MSRC) at https://msrc.microsoft.com/create-report.

If you prefer to submit without logging in, send email to secure@microsoft.com. If possible, encrypt your message with our PGP key; please download it from the Microsoft Security Response Center PGP Key page.

Please use https://g.co/vulnz to report security vulnerabilities.

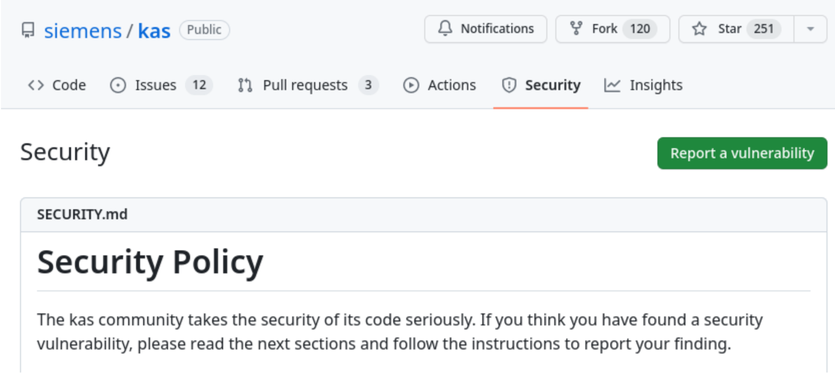
We use https://g.co/vulnz for our intake and triage. For valid issues we will do coordination and disclosure here on GitHub (including using a GitHub Security Advisory when necessary).

The Google Security Team will process your report within a day, and respond within a week (although it will depend on the severity of your report).

- **But then: https://github.com/google/oss-vulnerability-guide**
  - From simple to large projects
  - Templates, runbook etc.

Using git Forges as Infrastructure

- **GitHub**
  - Enable private vulnerability reporting (beta but working)
  - Patches can be developed on temporary private forks
  - Security advisories can be drafted privately and published later
  - Provides a calculator for the Common Vulnerability Scoring System (CVSS)
  - GitHub is a CVE Numbering Authority (CNA)
    - Once you accepted a report, you can request a CVE
    - @Siemens: also offered by ProductCERT
  - https://docs.github.com/en/code-security

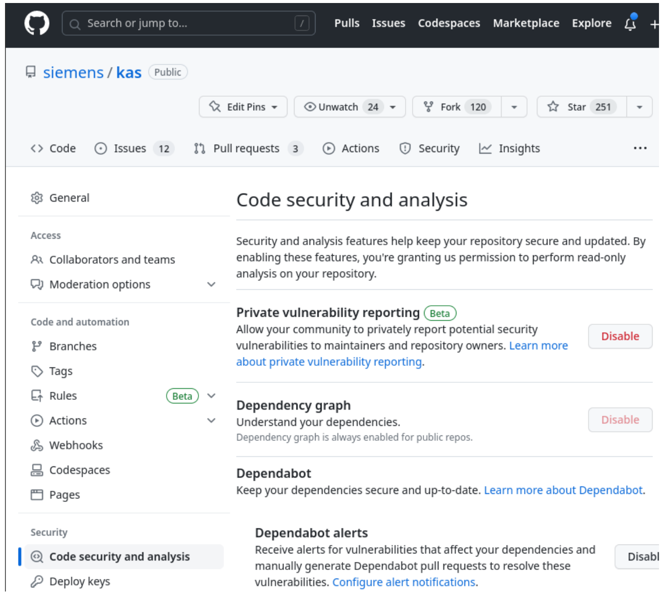


Security

SECURITY.md

Security Policy

The kas community takes the security of its code seriously. If you think you have found a security vulnerability, please read the next sections and follow the instructions to report your finding.



Code security and analysis

Security and analysis features help keep your repository secure and updated. By enabling these features, you're granting us permission to perform read-only analysis on your repository.

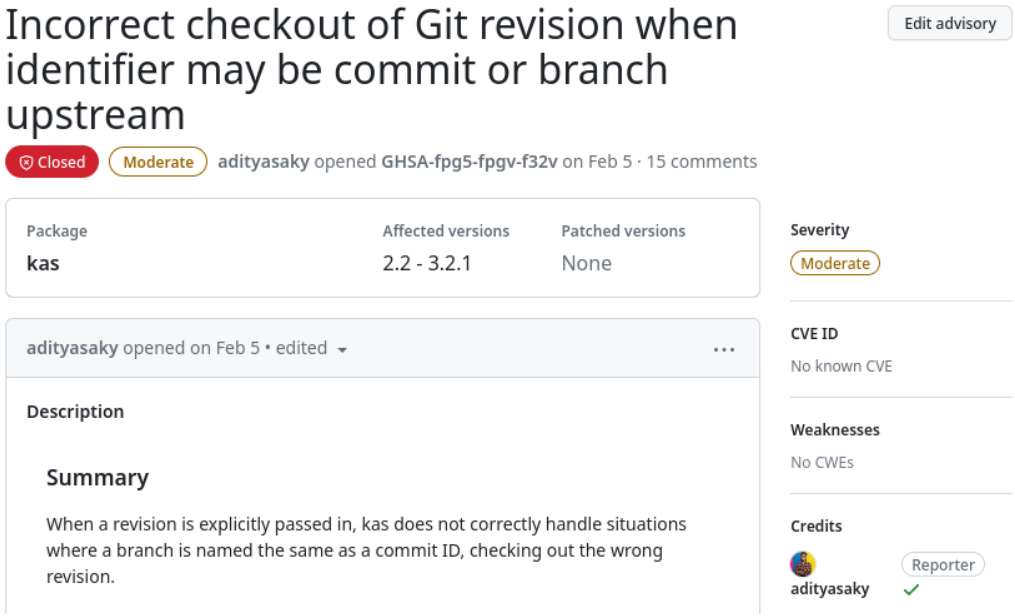
Private vulnerability reporting Beta
Allow your community to privately report potential security vulnerabilities to maintainers and repository owners. Learn more about private vulnerability reporting.

Dependency graph
Understand your dependencies.
Dependency graph is always enabled for public repos.

Dependabot
Keep your dependencies secure and up-to-date. Learn more about Dependabot.

Dependabot alerts
Receive alerts for vulnerabilities that affect your dependencies and manually generate Dependabot pull requests to resolve these vulnerabilities. Configure alert notifications.

impact alerts Beta



Incorrect checkout of Git revision when identifier may be commit or branch upstream

adityasaky opened GHSA-fpg5-fpgv-f32v on Feb 5 · 15 comments

Package	Affected versions	Patched versions
kas	2.2 - 3.2.1	None

Severity: Moderate

CVE ID: No known CVE

Weaknesses: No CWEs

Credits: adityasaky Reporter

adityasaky opened on Feb 5 • edited

Description

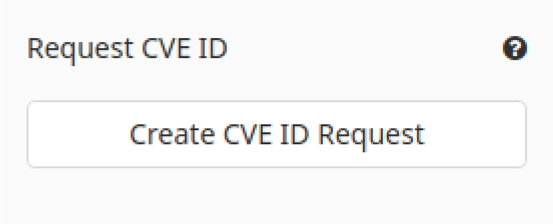
Summary

When a revision is explicitly passed in, kas does not correctly handle situations where a branch is named the same as a commit ID, checking out the wrong revision.

SIEMENS

Using git Forges as Infrastructure, Part II

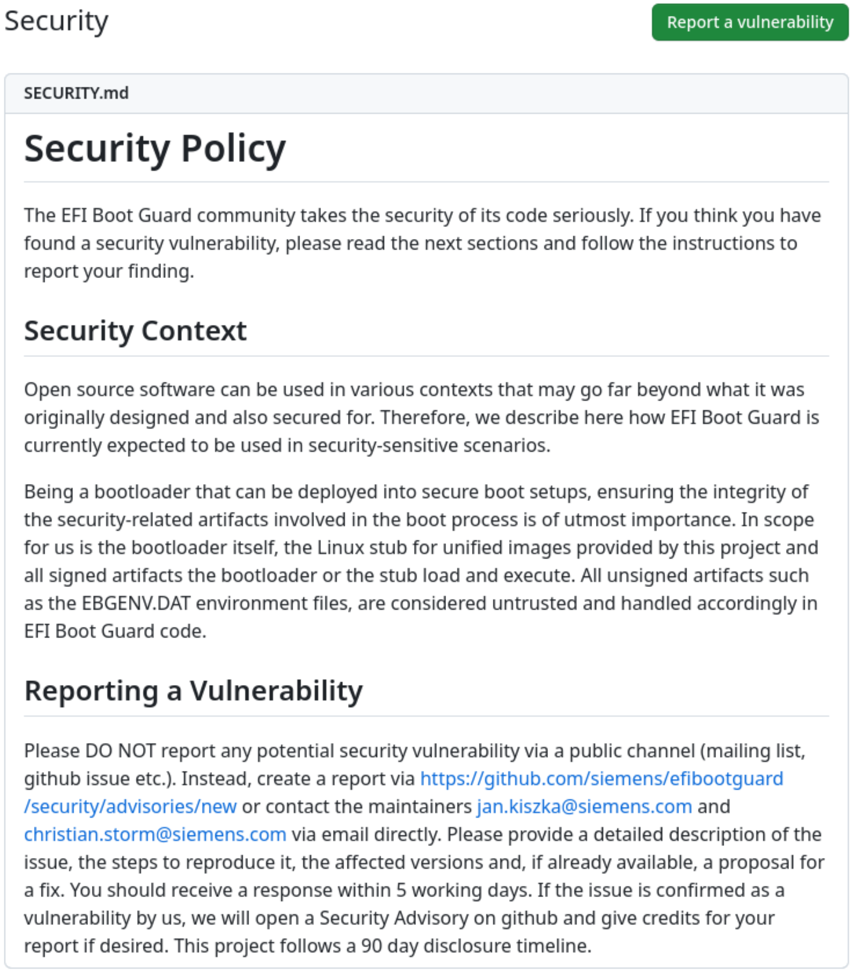
- **GitLab**
  - Vulnerability reporting via private issues
  - No equivalent to advisory (...or I was blind?)
  - Acts as CNA too (gitlab.com only)
  - Maintainer can request CVEs for private issue: https://docs.gitlab.com/ee/user/application_security/cve_id_request.html



- **Others**
  - Gitee? – didn't find anything (that I could parse)
  - Sourceforge?
  - ...?

SECURITY.md

- **Defines the project's security policy**
- **Picked up by GitHub prominently**
- **Suggested template (derived from Google)**
  - Introduction
  - Security context
    - What do you consider in-scope for the security of your project?
    - What is clearly not in-scope regarding security?
  - Reporting a vulnerability
    - Preferred contact channel
    - If email, consider a second contact
    - Define response time and disclosure timeline
- Examples
  - https://github.com/siemens/efibootguard/security
  - https://github.com/siemens/kas/security



Security Policy

The EFI Boot Guard community takes the security of its code seriously. If you think you have found a security vulnerability, please read the next sections and follow the instructions to report your finding.

Security Context

Open source software can be used in various contexts that may go far beyond what it was originally designed and also secured for. Therefore, we describe here how EFI Boot Guard is currently expected to be used in security-sensitive scenarios.

Being a bootloader that can be deployed into secure boot setups, ensuring the integrity of the security-related artifacts involved in the boot process is of utmost importance. In scope for us is the bootloader itself, the Linux stub for unified images provided by this project and all signed artifacts the bootloader or the stub load and execute. All unsigned artifacts such as the EBGENV.DAT environment files, are considered untrusted and handled accordingly in EFI Boot Guard code.

Reporting a Vulnerability

Please DO NOT report any potential security vulnerability via a public channel (mailing list, github issue etc.). Instead, create a report via https://github.com/siemens/efibootguard/security/advisories/new or contact the maintainers jan.kiszka@siemens.com and christian.storm@siemens.com via email directly. Please provide a detailed description of the issue, the steps to reproduce it, the affected versions and, if already available, a proposal for a fix. You should receive a response within 5 working days. If the issue is confirmed as a vulnerability by us, we will open a Security Advisory on github and give credits for your report if desired. This project follows a 90 day disclosure timeline.

Some General Maintainer Advises

- **If you received a report, acknowlege it soon**
- **No need to confirm validity prematurely!**
- **Assess, reproduce, discuss**
  - Consult other trusted core people
  - Involve reporter as well
- **Prepare a fix privately**
  - Don't be hectic!
  - Test carefully, even if fix is "obvious"
- **Look left and right for similar problems!**
  - Different parts of the code also affected?
  - New secure use cases?

Closing Thoughts

- **Check if your OSS project(s) should gain a SECURITY.md**
- **Do we want a default SECURITY.md for github.com/siemens?**
  - Would define a fallback if project has none
  - Would also add one for forks of other projects
- **Harden your project(s)**
  - Run security scanners for secrets, outdated dependencies etc.
  - Run code analysis tools
  - Secure your code access (work on trustworthy infrastructure, 2FA etc.)
  - Have a look at https://bestpractices.coreinfrastructure.org/en
    (but don't feel demotivated afterwards ;-))

Contact

Published by Siemens Technology

Jan Kiszka
Principal Key Expert
T RDA IOT
Otto-Hahn-Ring 6
81739 Munich
Germany

E-mail jan.kiszka@siemens.com