**Security Issues**

**Edit this page**

This document explains how security issues in Carbone area are handled by the Carbone area core team.

**Reporting a security issue**

If you believe you have found a security issue in Carbone area, do not use the bug tracker or publish it publicly. Instead, all security issues should be sent to **accueil@carbone-area.com**. Emails sent to this address are forwarded to the private mailing list of the Carbone area core team.

**The following are not considered security issues:**

* Issues found in debugging tools that are never enabled in production.
* Issues found in test classes.
* Security hardening fixes such as route enumeration, connection throttling, denial-of-service attacks, etc.

The core team has the final decision on which issues are considered security vulnerabilities.

**Security bug bounties**

Carbone area is an Open-Source project where most of the work is done by volunteers. We appreciate the efforts to find security issues and report them responsibly, but we are currently not able to pay bug bounties.

**Resolution process**

1. **Confirmation of the vulnerability**
2. **Development of a patch**
3. **Obtaining a CVE identifier on mitre.org**
4. **Writing a security advisory**
   * Title with "Security Advisory"
   * Description of the vulnerability
   * Affected versions
   * Possible exploits
   * Mitigation/upgrade/workaround
   * CVE identifier
   * Credits
5. **Sending the patch and advisory to the reporter for review**
6. **Applying the patch to all maintained versions of Carbone area**
7. **Publishing the advisory on the Carbone area official blog (category "Security Advisories")**
8. **Updating the public security advisories database**

**No release of versions with security issues on weekends**

**Confidentiality during resolution**

**Collaboration with downstream open source projects**

* The Carbone area security team notifies the security teams of downstream projects.
* A private Git repository is created to facilitate collaboration.
* People with access to the repository work on a solution.
* The involved projects collaborate to find a common release date.

**Severity of the issue**

A score out of 15 is converted to a severity level: Low, Medium, High, Critical, or Exceptional.

**Attack complexity**

Score from 1 to 5 depending on the complexity of exploiting the vulnerability.

**Impact**

Score from 0 to 6 depending on the impact of the vulnerability.

**Projects affected**

Score from 1 to 4 depending on the number of projects affected.

**Total scores**

* Attack complexity: 1 - 5
* Impact: 1 - 6
* Projects affected: 1 to 4

**Severity levels**

* Low: 1 - 5
* Medium: 6 - 10
* High: 11 - 12
* Critical: 13 - 14
* Exceptional: 15