Best Practices for Reporting Security Findings to Bizagi

Purpose

The purpose of this document is to provide a guide of best practices for reporting vulnerabilities or security findings identified during tests conducted by an external evaluator or hired by a client. Implementing these best practices will optimize the analysis time of the report and ensure an efficient and proportional response to the identified security risk.

Definition of Vulnerability

Understanding the definition of a vulnerability is crucial for effectively identifying and mitigating security risk. It also helps differentiate between a real security and best practice recommendations, allowing for the prioritization of actions and appropriate allocation of resources. A precise understanding of vulnerabilities enables informed decisions-making and the application of suitable security measures.

Bizagi applies the definition of vulnerability stablished by the CVE.org program. According to this definition, a vulnerability is: "An instance of one or more weaknesses in a product that can be exploited, causing a negative impact on confidentiality, integrity, or availability; a series of conditions or behaviours that allow for the violation of an explicit or implicit security policy."- Vulnerability Definition - CVE.org.

In other words, to determine if a finding is a vulnerability, it must be evaluated if there is an impact on one or more security attributes: confidentiality, integrity, or availability. To perform this evaluation, consider the following questions:

1. Confidentiality: Does the vulnerability allow an attacker to view or gain unauthorized access to sensitive data stored in Bizagi?
2. Integrity: Does the vulnerability allow an attacker to modify or destroy any type of information in Bizagi without authorization?
3. Availability: Does the vulnerability allow an attacker to cause a detriment or disrupt the service of any systems provided by Bizagi?

Bizagi recommends using the Common Vulnerability Score System (CVSS) in its versions 3.1 and 4.0. Using this system includes identifying the impact on these three attributes.

If the answer to these three questions determines that there is no impact, the finding would not be a vulnerability but could be a best practice in security. Unlike a vulnerability, the absence of a security best practice does not directly imply that an attacker will benefit from its exploitation. In security, a best practice could reduce the risk to which an application is exposed, a risk that would only be exploitable if a vulnerability is identified. Although there are well-known best practices in the market to improve application security, they cannot always be implemented in all aspects, as they greatly depend on the context in which an application is executed.

Examples of Security Best Practices vs Vulnerabilities

Consider the following examples to differentiate between a vulnerability and best practices:

Finding	Type	Justification
Restrictive definition in Content-Security-Policy	Best practice	The appropriate definition of the Content-Security-Policy helps to improve protections against code injection in front-end components. The inability to apply this restrictive policy is not considered a vulnerability since Cross-Site Scripting (XSS) is required to exploit a permissive Content –Security –Policy. Although risky options in Content-Security-Policy are widely known, Bizagi cannot remove them from all its applications due to the highly customizable nature of its products and services.
Security attributes in cookies	Best practice	Security attributes in cookies, such as Secure, HTTPOnly, SameSite=Strict, are used to protect the information in the cookies, Bizagi tries to implement these attributes in all defined cookies, but it cannot be implemented in all cases due to the nature of some applications. Cookies whose purpose is to manage languages, customize user interface themes, etc., cannot have the HTTPOnly attribute because they need to be read by the front-end. They also do not require the Secure attribute because they do not handle sensitive information.
Updating a component without known vulnerabilities.	Best practice	If an outdated component is identified without any known vulnerability impacts to date, it is not considered a vulnerability.

		Although the component is not the latest version, without a registered vulnerability, it cannot be determined if it is insecure, so updating it is considered a best practice.
SQL injection	Vulnerability	This vulnerability would allow an attacker to manipulate SQL queries through an application. It is a good example of a vulnerability as it *compromises the integrity and confidentiality* of the information stored by an application.
Cross-Site Scripting (XSS)	Vulnerability	Allows injecting malicious instructions into web pages, affecting the browser of a legitimate application user. This affects the *integrity and confidentiality* of the user's data.
Use Unencrypted channels	Vulnerability	Not securing a communication channel could compromise the *confidentiality* of data during transmission, potentially leading to unauthorized information leakage.

Considerations Before Reporting

Apply the following considerations begore reporting a security finding to ensure an efficient response from the Bizagi team.

1. Verify if your finding meets the definition of vulnerability. Before reporting a security finding, ensure it meets the criteria provided in this article. Bizagi prioritizes vulnerabilities based on the severity and impact provided in the report. Best practices are also received and addressed but do not require immediate attention and may be applied in future versions.
2. Read the vulnerability disclosure policy. The vulnerability disclosure policy establishes communication channels whit Bizagi, as well as the terms and conditions when reporting a security finding. The policy is available at: Vulnerability Disclosure Policy | Bizagi
3. Avoid sending security tool reports without context and prior analysis. Security tools often generate false positives. To prevent long response time from our team, try to report only security findings recognized as real vulnerabilities. Provide context for each of the reported findings. Without proper context, the Bizagi support team will have to analyse the security report on their own, which takes much more time and can delay the resolution of associated vulnerabilities.
4. Divide complete security report into specific and individual cases. Dividing the report allows for separate responses to findings, improving the traceability of applied actions and facilitating follow-up on remediation plans.
5. Assing a coherent severity to the identified finding. Use a standard methodology such as CVSS in its versions 3.1 or 4.0 to identify the severity of a finding. This system will allow you to assess the impact on confidentially integrity, and availability. If you want to use another system, try to explain the severity of the finding in its description.
6. Include the necessary information for each finding. For an effective report, ensure to include the following points in your report.

- Description: Briefly describe the finding, including associated threats and risk. The description should answer: How does an attacker benefit from exploiting this vulnerability? Include the impact description in terms of Confidentiality, Integrity and Availability.
- Type of vulnerability: Example: SQL Injection, Cross – Site Scripting (XSS), OS Command Injection.
- Severity: Using the CVSS Base in version 3.1 or 4.0
- Related CVE entry: If an outdated component with known vulnerabilities is identified.
- URL of the affected application or version of the component where the finding is identified.
- Proof of concept: With the necessary step-by-step evidence to reproduce the scenario. Try to attach images or videos describing the exploitation steps.
- Special configurations or preconditions to reproduce the scenario: Specially if a specific Bizagi model is needed to reproduce it.
- Impact: Describing the benefit an attacker would have by exploiting this vulnerability
- Mitigations and remediations recommendations.