BIECO

Building Trust in Ecosystems
and Ecosystem Components

Best Practice Handbook

bieco.org

BIECO Classification of Security Best Practices

The following classification outlines essential security best practices that a system should incorporate. These practices are grouped into the following categories:

- **Authentication:** This involves the verification of claimed identities.

- **Integrity:** Ensuring the accuracy and completeness of data throughout its lifecycle. This prevents unauthorized or unnoticed data modifications.

- **Non-repudiation:** This pertains to honoring contractual obligations. It also prevents transaction parties from denying sending or receiving a transaction.

- **Confidentiality:** This prevents unauthorized individuals, entities, or processes from accessing or disclosing information. While related to privacy, the two terms have distinct meanings. Confidentiality safeguards data from unauthorized access.

- **Availability:** This means the computing systems used to store and process the information, the security controls used to protect it, and the communication channels used to access it must be functioning correctly.

- **Authorization:** This means that there are security mechanisms to determine access levels or user/client privileges related to system resources, including files, services, computer programs, data, and application features.

To enhance the definition of the claims, we also consider the impact dimension that can happen as a consequence of not fulfilling a specific security claim, such as privacy leakage or a safety issue resulting in human life losses. We consider the following categorization for impact:

- **Safety:** Unwanted or unauthorized interference with the system or communications that may impact the safe operation of the system.
- **Financial:** Unwanted or unauthorized commercial transactions, or access to the system that may imply theft of the system, intellectual property infringement, damage to manufacturer reputation, or warranty fraud, among others.
- **Operational:** Unwanted or unauthorized interference with the system or communications that may impact the operational performance of the system (without affecting physical safety).
- **Privacy and Legislation:** Unwanted or unauthorized acquisition of data relating to system activity, user identity data, or system design and implementation, and non-compliance with relevant legislations.

This classification framework aims to provide a comprehensive understanding of security best practices and their potential consequences.

Authentication claims

This claims group is focused on the verification of the necessary access control mechanism to protect the access to the Target of Evaluation (TOE), its updates, interfaces and services. Moreover, this group also checks the strength of the authentication mechanisms used and its resistance against well-known attacks such as brute force or side channel attacks.

We also included general claims that could affect authentication, such as the presence of vulnerabilities in the used libraries or the strength of dependent mechanisms (e.g., random bit generators).

The following claims are associated with the authentication group:

- Update software files should be authenticated.
- The exchanged messages in the communication should be integrity protected
- Automatically generated passwords should be unique.
- Passwords should avoid common patterns.
- Passwords are not obviously linked to public information.
- Passwords should be strong in terms of complexity.
- The changes of the authentication values for user authentication are successful.
- Access to device functionality via a network interface in the initialized state should only be possible after authentication on that interface.
- The system should have a mechanism available which makes brute- force attacks on authorization mechanisms via network interfaces impracticable.

Authentication claims

- Authentication mechanisms must use strong passwords.
- Connections to remote services, interfaces, and end-points should be cryptographically authenticated.
- The software should not use unsafe libraries that contain vulnerabilities.
- Protocols and libraries used by the system are updated.
- Authentication protocols should be secure, using recommended algorithms.
- Authenticated sessions should expire, and a new re-authentication required.
- Random bit generators should be strong enough.
- Authentication algorithms should avoid channel side attack.
- The system shall enforce a limit of consecutive invalid login attempts during a time period.
- The system shall notify, upon successful logon, of the date and time of the last logon and the number of unsuccessful logon attempts since the last successful logon.
- The system shall uniquely identify and authenticate users.
- The system shall uniquely identify and authenticate a defined list of devices before establishing a connection.
- The system shall lock the session after a configurable time period of inactivity.
- The system shall terminate a remote session at the end of the session or after a period of inactivity.

Authorization claims

This group of claims focuses on verifying the necessary mechanisms to ensure that only authorized users/entities can access the services and data of the TOE. Additionally, this group examines the strength of the authorization mechanisms used, along with their compliance with the XACML standard. One important aspect covered within this group is the metrics related to authorized access to private data stored and shared with the TOE, in accordance with the GDPR regulations.

As before, we also included general claims that could affect authorization, such as the presence of vulnerabilities in the used libraries or the strength of dependent mechanisms (e.g., random bit generators).

The following claims are associated with the authorization group:

- All unused network interfaces shall be disabled.
- Lawfulness of processing of personal data.
- Personal data must be processed for a specific purpose.
- The system should allow data subject to access its personal data
- The adopted XACML-based authorization mechanism (PDP) must implement the mandatory functionalities of the XACML standard specification language.
- The access control mechanism (PDP) that evaluates the authorization requests against a policy must correctly implement
- the policy.
- The adopted XACML access control policy must be correct with respect to a specification (model) of the access control rights.

Authorization claims

- Endpoints should only run applications or services whose TCP or UDP ports are described in the MUD profile. Unnecessary interfaces, and services should be disabled.
- A MUD file should be provided in accordance with MUD RFC.
- The software should not use unsafe libraries that contain vulnerabilities.
- Protocols and libraries used by the system are updated.
- Random bit generators should be strong enough.
- The system should allow data subject to modify its personal data.
- The system should allow data subject to delete permanently personal data concerning it.
- The system should allow data subject to withdraw its given consent
- The system shall enforce assigned authorizations for controlling the flow of information within the system and from interconnected systems.
- The system shall monitor events to detect attacks, unauthorized activities or conditions, and non-malicious errors.
- The system shall terminate a remote session at the end of the session or after a period of inactivity.
- The system shall provide a proof of notice requirements for an explicit consent receipt for demonstrating compliance with the GDPR.

Integrity claims

This claims group is focused on the verification of the necessary mechanisms to ensure that all the data stored and exchanged in any communications with the TOE is protected against modifications.

As before, we also included general claims that could affect integrity, such as the presence of vulnerabilities in the used libraries or the strength of dependent mechanisms (e.g., random bit generators).

The following claims are associated with the integrity group:

- Update software files should be integrity protected.
- The exchanged messages in the communication should be integrity protected.
- Sensitive security parameters exchanged during the communication for the establishment of a secure association should be integrity protected.
- Stored sensitive security parameters should be integrity protected.
- Integrity mechanisms must be strong.
- The source code must not contain SQL injection vulnerabilities.
- The source code must not contain command injection vulnerabilities.
- The source code must not contain code injection vulnerabilities.
- The source code must not contain path traversal vulnerabilities.
- The source code must not use components with known vulnerabilities.

Integrity claims

- Automatic updates should not change the network protocol interfaces in any way that is incompatible with previous versions.
- The software should not use unsafe libraries that contain vulnerabilities.
- Protocols and libraries used by the system are updated.
- Random bit generators should be strong enough
- The system shall implement mechanisms of protection from malicious code manipulation.
- The system shall update protection mechanisms whenever new releases are available.
- The system shall prevent anyone from circumventing malicious code protection mechanisms.
- The system shall execute a fail-safe procedure upon the loss of communications with other systems.
- The system shall uniquely identify and authenticate users.
- The system shall isolate security functions from non-security functions.
- The system shall separate user functionalities from management functionalities.
- The system shall monitor events to detect attacks, unauthorized activities or conditions, and non-malicious errors.
- The system shall prevent messages from being received from external users or systems.
- The system shall operate in a degraded mode during a DoS event.
- The system shall limit the use of resources by security functions to prevent resource exhaustion.

Availability claims

This claims group is focused on the verification of the necessary mechanisms to ensure that the system is permanently in operation and no external faults can alter its normal functionality. A failure in the availability of the system can lead to significant monetary and operational losses.

As before, we also included general claims that could affect availability, such as the presence of vulnerabilities in the used libraries or the strength of dependent mechanisms (e.g., random bit generators).

The following claims are associated with the availability group:

- The update mechanism shall prevent downgrade.
- Resistance to DoS attacks.
- Data input validation.
- The source code must not contain SQL injection vulnerabilities.
- The source code must not contain command injection vulnerabilities
- The source code must not contain code injection vulnerabilities.
- The source code must not contain path traversal vulnerabilities.
- The source code must not use components with known vulnerabilities.
- Automatic updates should not change the network protocol interfaces in any way that is incompatible with previous versions.
- The software should not use unsafe libraries that contain vulnerabilities.

Availability claims

- Device should remain operating and locally functional in the case of a lost network connection.
- Protocols and libraries used by the system are updated.
- Random bit generators should be strong enough.
- System should work in case of power outage.
- The system shall update protection mechanisms whenever new releases are available.
- The system shall prevent anyone from circumventing malicious code protection mechanisms.
- The system shall execute a fail-safe procedure upon the loss of communications with other systems.
- The system shall set outputs to a predetermined state if normal operation cannot be maintained as a result of an attack.
- The system shall operate in a degraded mode during a DoS event.
- The system shall limit the use of resources by security functions to prevent resource exhaustion.

Confidentiality claims

This claims group is focused on verifying that any data or message exchanged or stored in the TOE is protected against prying attackers and sniffers. This not only guarantees data privacy, but also prevents an attacker from obtaining additional information useful to plan a future attack.

As before, we also included general claims that could affect confidentiality, such as the presence of vulnerabilities in the used libraries or the strength of dependent mechanisms (e.g., random bit generators). We also include claims derived from the currently applicable EU legal framework (i.e., the GDPR) so as to guarantee lawfulness processing of personal data.

The following claims are associated with the confidentiality group:

- Update software files should be encrypted and be transmitted using encryption.
- Update software files should be encrypted using strong keys and algorithms.
- Automatically generated passwords should be unique.
- Passwords should avoid common patterns.
- Passwords are not obviously linked to public information.
- Passwords should be strong in terms of complexity.
- Stored critical security parameters should be ciphered.
- Ciphered communications should use strong algorithms.
- Critical security parameters should be encrypted in transit, with such encryption appropriate.
- The confidentiality of personal data transiting between a device and a service, especially associated services, should be protected, with best practice cryptography.

Confidentiality claims

- Data Communications should be ciphered.
- Lawfulness of processing of personal data.
- Personal data must be processed for a specific purpose.
- The system should allow data subject to access its personal data.
- The source code must not contain SQL injection vulnerabilities.
- The source code must not contain command injection vulnerabilities
- The source code must not contain code injection vulnerabilities.
- The source code must not contain path traversal vulnerabilities.
- The source code must not use components with known vulnerabilities.
- The software should not use unsafe libraries that could derive on vulnerabilities.
- Protocols and libraries used by the system are updated.
- Random bit generators should be strong enough.
- The system should allow data subject to modify its personal data.
- The system should allow data subject to delete permanently personal data concerning it.
- The system should allow data subject to withdraw its given consent
- The system shall enforce assigned authorizations for controlling the flow of information within the system and from interconnected systems.
- The system shall prevent messages from being received from external users or systems.

Non-repudiation claims

This claims group focuses on verifying that transactions are properly registered and that attempts to erase traces of malicious activities from this registry are not possible. It also involves aspects related to explicit consent (as defined in the GDPR), ensuring that both parties are aware of and recorded under the established conditions.

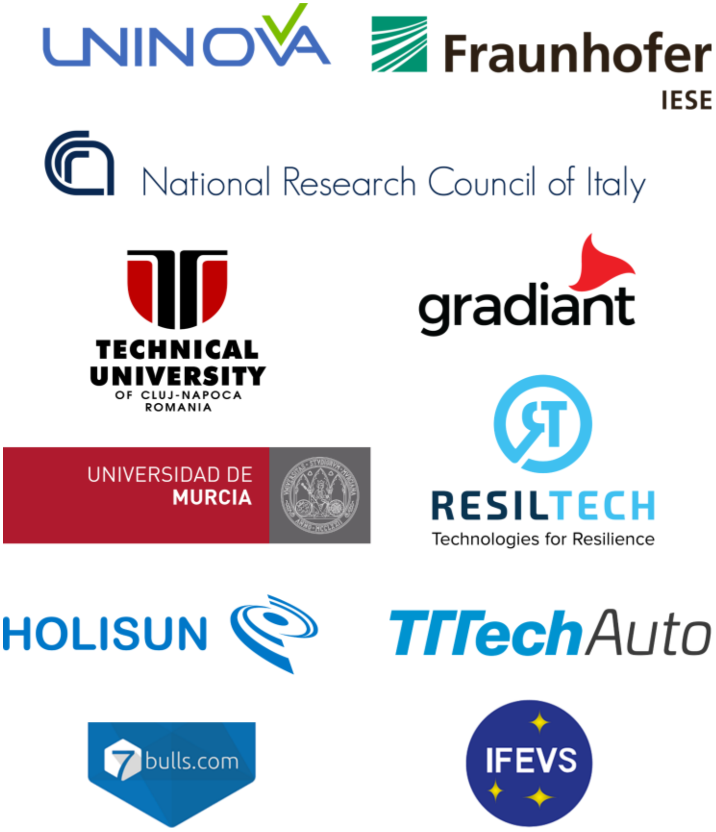
As before, we have also included general claims that could impact confidentiality, such as the presence of vulnerabilities in the libraries used or the strength of dependent mechanisms (e.g., random bit generators).

The following claims are associated with the non-repudiation group:

- The software should not use unsafe libraries that contain vulnerabilities.
- Protocols and libraries used by the system are updated.
- Random bit generators should be strong enough.
- The system shall uniquely identify and authenticate users.
- The system shall uniquely identify and authenticate a defined list of devices before establishing a connection.
- The system shall monitor events to detect attacks, unauthorized activities or conditions, and non-malicious errors.
- The system shall lock the session after a configurable time period of inactivity.
- Logs should be protected against removal.
- The system shall provide a proof of notice requirements for an explicit consent receipt for demonstrating compliance with the GDPR.

BIECO
Building Trust in Ecosystems
and Ecosystem Components

The BIECO consortium



Building Trust in Ecosystems and Ecosystem Components

This project has received founding from the European Union’s Horizon 2020 Research and Innovation Program under Grant agreement No. 952702.