Wipfli Product Development Secure Coding Practices

Purpose

This document is intended to provide, in general, the high-level coding practices implemented by Wipfli in the development of licensed products available for subscription purchase ("Wipfli SaaS software products"). As a guide, the Open Web Application Security Project ("OWASP") has been leveraged for contextual categories of practices defined here within. The definitions documented represent our commercially reasonable efforts at building and supporting secure products available for purchase on the open market. This is not a guarantee or legally binding document but rather represents customary practices in place to help mitigate most common software vulnerabilities.

All Wipfli SaaS software products ("Wipfli products") shall be subject to the Wipfli Software License & Hardware Purchase Agreement which can be referenced here.

Input Validation

Wipfli products are developed within secure environments and deployed within customer owned servers or cloud instances. We build input validation into any data elements required by our products from a user and leverage official publisher API's and Web Services where appropriate to ensure the proper business logic is applied to any data entering a customer system, rather than any direct SQL or database direct access. This insures proper field level validations, workflows, and business rules are applied to all data inputs.

Output Encoding

Wipfli products are not intended for general use but rather subscribed by individual customers and deployed within their secure environments. There is no data that would be returned to the local environment or user that originated outside the applications trust boundary. Therefore, our output encoding uses the native application's encoding standard (e.g. typically UTF-8).

Authentication and Password Management

Wipfli products require user authentication at the source application or core accounting system level before accessed. Our products do not store any passwords or usernames and use only HTTP POST requests for web API authentication credentials. Since our products rely on access and permissions managed by the source application or core accounting system, all authentication controls for these source systems must be fully vetted by the customer and end users. For all web API calls, we rely entirely on tokenized OAuth authentication methods as available by the platform vendor.

Session Management

Wipfli products leverage the appropriate publishers session management controls when making any API based sessions for data exchange. Our products generate a new API session from an existing session before running further API calls.

Access Control

Wipfli products use only trusted system objects for making access authorization decisions. These are specifically provided by the source application or core accounting system. Our products leverage the publishers permissions and roles to contain access within least privilege principles.

Cryptographic Practices

Wipfli products do not have necessity or data scope to require cryptographic practices.

Error Handling and Logging

Wipfli products provide generic error messages and do not disclose sensitive information in error responses our output logging.

Data Protection

Wipfli products do not include sensitive information in API or Web Service communication. In addition, Wipfli products do not store passwords, connection strings, or other sensitive information in clear text.

Communication Security

Wipfli products use the latest TLS protocol (e.g. TLS 1.2 as of right now) for all web API calls. Failed TLS connections do not fall back to an insecure connection. Where available and applicable, Wipfli products have been certified by the source application or core accounting system publisher for our usage of their web APIs.

System Configuration

Wipfli routinely updates and patches all systems involved in the secure coding of our Wipfli products. Components and service accounts leveraged are restricted to the least privileges possible. Wipfli products leverage best of breed cloud PaaS solutions (e.g. Heroku AWS and Azure) where appropriate and core systems data is routinely backed up.

Database Security

Wipfli products leverage abstractions like LING-to-SQL where appropriate and parameterized SQL whenever direct SQL is most appropriate. Wipfli products do not leverage SQL string building for any database access.

File Management

Wipfli products do not provide any type of file storage or management. PDF or image documents transferred through our products are stored within the source application or core accounting system and thus those publishers would be providing secure File Management practices.

Memory Management

Wipfli products use commercially reasonable approaches for memory management.

General Coding Practices

Wipfli products are developed with commercially reasonable coding best practices. This includes the use of best of breed source control systems such as git and BitBucket. The small, discrete amount of customer data collection needed for our products has been documented and can be delivered to individual customers upon request.