PaaSword Reference Architecture

27. October 2015

Cloud computing has evolved from a promising concept to one of the fastest growing segments of the IT industry. However, many businesses and individuals continue to view cloud computing as a technology that risks exposing their data to unauthorized users. To this end, the responsibility of the developers has been significantly increased since modern applications should be built using a security-by-design paradigm. Security-by-design implies that specific best practices regarding the secure storage and the secure interaction with an application's data should be enforced in an automated way. To do so, PaaSword project aims to deliver a framework according to which these best practices can be enforced seamlessly using source code annotations. These annotations can be used in order to drive specific business logic in a seamless way during run-time.

The scope of this deliverable is to provide the reference architecture of the PaaSword framework. The reference architecture aims to describe the design-time and run-time components that are required in order for the security-by-design concept to be realized. These components have been designed based on the functional and non-functional requirements that have been raised by the PaaSword end-users. The cornerstone component of the architecture is the PaaSword Context Model that conceptualizes the attributes of an application's operational environment which will be taken under consideration during run-time in order to perform security management of persisted data and policy enforcement of end-users.

The Context Model is 'translated' into typesafe libraries that can be used by developers in order to produce PaaSword-enabled applications. Such applications can be deployed in a PaaSword container which is the operational environment that interprets the annotations in proper security policies. These policies refer either to the application end-user and his/her ability to interact with the application or to the transparent encryption that will be performed in the underlying database.

The transparent encryption reassures that data that exists in outsourced databases cannot be processed by malicious users. However, beyond the symmetric encryption algorithm that has to be used in a transparent way the encryption key management is a crucial aspect. The key management can be performed in various ways. Therefore, PaaSword will support several transparent encryption policies based on the application requirements. As it will be presented, there is no holy grail policy regarding these policies since security level, complexity and efficiency contradict to each other.

The current deliverable elaborates on the aforementioned components in a coarse grained view since the normative specification of each component will be delivered in the frame of Work Packages 2, 3 and 4. However, specific implementation guidelines are thoroughly discussed.