Security Properties of Software Components

Khaled Khan, Jun Han, and Yuliang Zheng

Peninsula School of Computing and Information Technology.
Monash University
McMahons Road, Frankston
VIC 3199 Australia.
Fax: +61-3-99044124
{khaled,jhan,yuliang}@mars.pscit.monash.edu.au

Abstract. This paper classifies security properties of software components into two broad categories: (1) non-functional security (NFS) properties, and (2) properties as security function (SF). Non-functional security properties are codified and embedded with the component functionality, whereas, properties as security functions are employed as external protection to the component. In most cases, users may add additional external protection to the binary form of the component. This classification could be used to determine how much the overall security of the component is dependent on the non-functional security properties of the component and to what extent the additional external protections are required in order to use the component in their specific application environment.

Keywords. Software component, Security properties, Component functionality

1 Motivation

Software component security is concerned with the protection of the resources, data, and computational properties of individual components and the enclosing system. The security of software component is very much dependent on the role that the component plays in a specific application environment. Software components need to be customised with the application environment where it is deployed. In most cases, third-party software components need external protection employed by the users to meet their specific application requirements. Due to the binary representation of the component, software composers are not able to modify the security properties that are embedded at the implementation level. We believe that overall security properties of software components should be classified into two categories, and this information should be available for users' inspection before a candidate component is selected.

There is a growing concern on the issue of characterising the component security properties in recent days as expressed in [2], [5], [6], [7], [8]. The purpose of this paper is to identify various sources of security features that are codified with the component functionality. Most of the research conducted in recent days...
related to component security are involved in detecting and removing the security flaws of components. We believe that security of component should be considered differently than the case of application systems because of the distributed usage of the component. In this paper, we are not going to propose any new security technique or assessment, rather we define different types of security properties that are placed various ways enforcing certain security objective of the component.

2 Classes of Component Security Properties

In this section, we define two distinguished classes of security properties related to software components such as properties as security function (SF) and non-functional security (NFS) properties.

Properties as security functions (SF) are those that provide secure encasing to the components, protecting the component from any security threat from the outside of its boundary. External security features can be added to components as functions. We call them security functions (SF). On the other hand, non-functional security (NFS) properties are inherited in the internal implementation of the component, that is, some security features are codified with the functionality that the component provides. We term these features as non-functional security properties (NFS). Usually, NFS properties are some security enforcement mechanisms that are already embedded in various forms with the functionality of the component at the implementation level. NFS properties are, in fact, the implementation of highly abstract security objectives that are intended to counter certain security threats. NFS properties are attached with various aspects of the component functionality in different layers of implementation, each representing a specific level of abstraction to achieve certain security objective. Whereas, SFs are considered some kind of external protection mechanisms of components which are not directly related to the component functionality. SFs can be designed and added to the existing component. For example, a wrapper can be written to protect the component from a potential security threat. The authentication of component origin and identity is considered as a SF. This type of SF protects the enclosing system from being assembled with an unauthorised component. Particularly in a dynamic assembly scenario, a target component may be located in a remote server whose identities may or may not be authenticated.

NFS properties are embedded with the component functionality to prevent the threat of usage violation. A component may employ certain NFS properties to guard its sensitive data and functionality from being violated by other unauthorised entities. The NFS properties embedded with the component’s functionality may have substantial impact on the entire security mechanism of the composed system. It is not enough to make a system secure by adding SF such as encryption technology or secure protocols for data transmission to the component if the implementation of the component has security flaws. A great effort in de-