4.1.1.3 Runtime Platforms

Return to Platforms

About

Return to Top

A **Runtime Platform** is the software components that are required to run application software that are not part of the Application itself, Hardware Platform or Operating Systems Platform.

In general, a computation is a sequence of steps/events performed by the system or one of the activities within the system. The computation is performed by the code, which is supported by other components of the system. Code provides the constraints to computations and therefore, determines which computations can occur. For example, the computation is determined by the control flow relationships between individual activities, and further by the data flow relationships describing how individual activities act as producers and consumers of data. These relationships are the constraints for the computation

In order to analyze a computation, all system components have to be considered, including the so-called application code, but also the runtime platform and all runtime services, as some of the key control and data flow relationships are provided by the runtime platform, as the computation flows through the application code into the runtime platform and services and back to the application code. Application code alone in most cases does not provide an adequate picture of the computation, as some segments of the flow are determined by the runtime platform and are not visible in the application code. For example, while a large number of control flow relationships between different activities in the application code are explicit (such as statements in a sequence, or calls from a statement to another procedure), some control flow relations are not visible in the code, including the so-called callbacks, where the application code registers a certain activity with the runtime platform (for example, an event handler or an interrupt handler) and it is the runtime platform that initiates the activity. Without the knowledge of such implicit relationships, the system knowledge is not complete at a very fundamental level, leading to incomplete coverage of code analysis by vulnerability detection tools, and subsequently, to false negative and false positive report findings.

Mansourov, Nikolai, Campara, Djenana; 2011
https://www.sciencedirect.com/topics/computer-science/runtime-platform

From:
https://omgwiki.org/dido/ - **DIDO Wiki**

Permanent link:
https://omgwiki.org/dido/doku.php?id=dido:public:ra:1.4_req:1_func:platform:sw_arch&rev=1607618258

Last update: **2020/12/10 11:37**