Managing Changing Security Requirements using Release Planning

The Open University — Centre for Research in Computing

Saad Bin Saleem, Dr. Yijun Yu, Professor Bashar Nuseibeh and Professor Anne De Roeck

Research Context

In recent years, **the growing number of security attacks** has increased the need to handle security of the **evolving software systems**; in such systems the inevitable **security changes** can cause **violation of the existing security requirements**. Therefore, **managing changing security requirements** is difficult in order to preserve or enhance security of the evolving systems in a controlled way. **Release Planning (RP)** provides a controlled way to manage changes in the software as a structured process. However, **RP does not provide concrete measure for preserving/enhancing security in terms of managing a change** in related security and the other functions named as security dependency. Therefore, I propose to extend and apply RP to manage security of the evolving systems by managing the security dependencies.
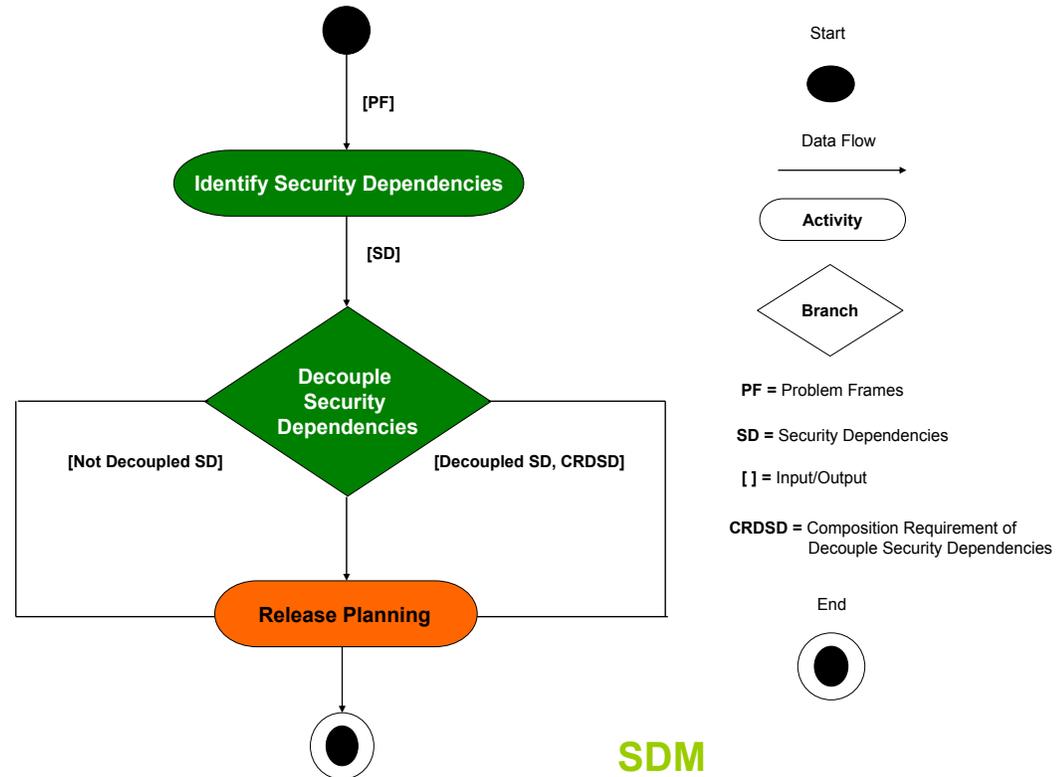
Research Question: *How can security dependencies be managed to accommodate changes to security requirements through release planning in order to keep satisfying security requirements of the evolving systems?*

Security Dependency Management Approach (SDM)

SDM is developed in order to address the problem of managing changing security functions related with the other functions as shown in the figure. The problem diagrams are provided as input to the approach; the following are two steps of approach before allocating a refined input for planning the releases.
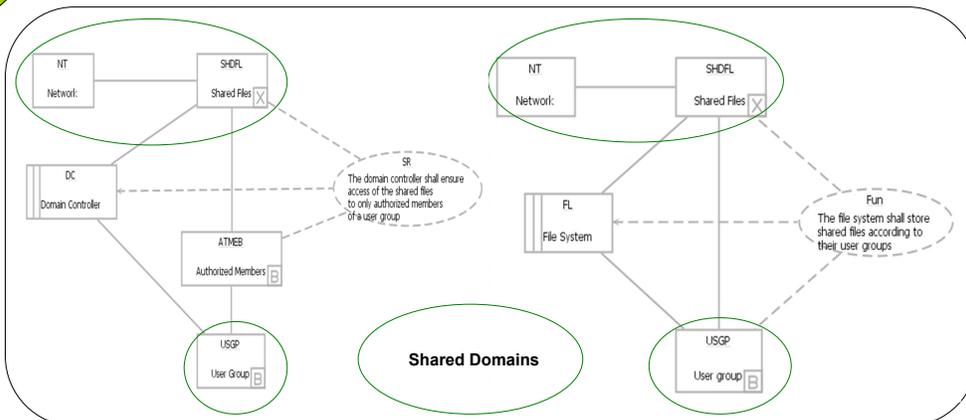
Identify Security Dependencies: A security dependency related through shared domains is identified based on the modelled requirements using the Problem Frames approach.

Decouple Security Dependencies: A decision is made about whether such a security dependency can be decoupled or needs to be to coupled to release planning; if it can be decoupled, a composition requirement of a decoupled security dependency is derived.
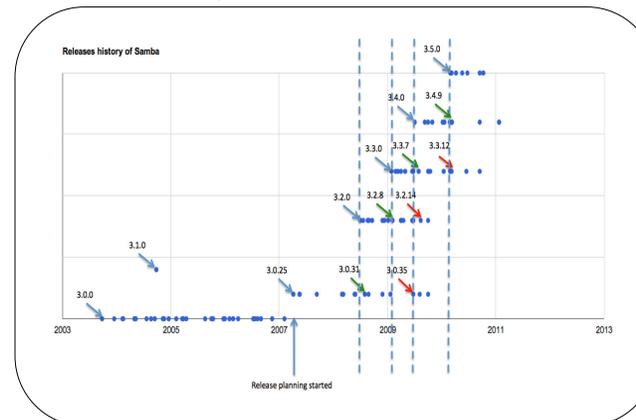
SDM

Start
Data Flow
Activity
Branch

PF = Problem Frames
SD = Security Dependencies
[ ] = Input/Output
CRDSD = Composition Requirement of Decouple Security Dependencies
End

[PF] → **Identify Security Dependencies** → [SD] → **Decouple Security Dependencies** → [Not Decoupled SD] / [Decoupled SD, CRDSD] → **Release Planning**

An Example

Shared Domains

Case Study

Releases history of Samba

References

1. A. Nhlabatsi, B.Nuseibeh and Y. Yu, Security requirements engineering for evolving software systems: A survey. Journal of Secure Software Engineering, 1(1):54_73, 2009.

2. S. B. Saleem, Y.Yu, B.Nuseibeh, and A. D. Roeck. "An Evaluation of the ReleasePlanner on the Treatment of Security and Evolving Requirements," *TR-CRC-OU-2010-19*, 2010