Unit 7: Introduction to Operating Systems

E-Portfolio Component:
Faceted Data

Task:

Read Schmitz et al (2016) article about faceted data.

- Do you think this is a good approach to protect systems from data leakage? What are the pros and cons?

Create a basic outline design of how you would create such a system in Python.

Answers:

Question 1

Based on my analysis of Schmitz et al.'s (2016) work on faceted data, this approach offers several significant advantages while also having some limitations for protecting systems from data leakage.

The key strength of faceted values is their ability to simulate secure multi-execution within a single process, providing strong information flow guarantees while improving performance compared to running multiple separate executions (Schmitz et al., 2016). The approach effectively handles both explicit and implicit information flows, preventing direct data leaks as well as indirect leaks through control flow. The implementation as a Haskell library rather than a language modification makes it more practical to adopt, as it doesn't require changes to the runtime environment. The authors demonstrate that their approach guarantees termination-insensitive non-interference, meaning private inputs cannot influence public outputs.

However, there are some limitations to consider. As noted in prior work by Zanarini et al. (2013, cited in Schmitz et al., 2016), secure multi-execution approaches can alter the behavior of programs that violate non-interference, potentially introducing bugs that are difficult to analyse. While faceted evaluation reduces redundant calculations compared to full multi-execution, there is still performance overhead from maintaining and processing multiple views of data. Additionally, the approach requires developers to explicitly mark sensitive information and properly configure security policies - incorrect marking or policy specification could still lead to vulnerabilities.

Despite these limitations, the faceted data approach represents a promising direction for information flow security, particularly given its ability to be implemented as a library and its strong theoretical guarantees. The trade-off between security and performance appears reasonable for many applications where protecting sensitive data is critical.

References:

Schmitz, T., Rhodes, D., Austin, T.H., Knowles, K. and Flanagan, C., 2016. Faceted Dynamic Information Flow via Control and Data Monads. In: F. Piessens and L. Viganò, eds. POST 2016. Berlin: Springer, pp.3-23.

Zanarini, D., Jaskelioff, M. and Russo, A., 2013. Precise enforcement of confidentiality for reactive systems. In: Computer Security Foundations Symposium. IEEE, pp.18-32.