**Description and Prototype**:

We would like to detect the following secure decisions:

1. Access Control, which is to “decide whether access to a protected resource should be granted or denied”.It involves confidentiality, authentication, authorization, privacy of the users, and the validation of resources.
2. Threat and Risk Assessment, which is to “make judgements based on knowledge, vulnerabilities, threats, risks, impacts and probability”.

We solved the problem in the following steps:

1. Looking into the source code of the system from Tomcat 5.5 to 9.0 for the particular secure decisions. Analyzing implementation and extracting features (like the keywords) in the module name or source code.
2. Filtering the modules through the list of extracted features. For some techniques like ARC, it only examines “document level concerns”, while we go deeper into doing analysis in a lower level of implementation, which means we not only look for the features in the filename or documentation, but also its source codes. We assign the weight for the features appeared in filenames or class names of the source codes, with the filenames have more weight than the class names appeared in source codes.
3. By making use of the filenames for component modules generated from the codes of recovery techniques, we read these files, and then filter and cluster all the package and class names as the components of the security decision.
4. Visualizing the results. We put all the components from the generated clustering, and denote their relation with respect to different secure decisions.

What we have done:

By implementing the extension of the techniques, we can cluster all the element related to the particular decisions instead of roughly classify the components in ACDC technique, and visualize our solution of components in the level of class names to make it easier to understand.

Our modification:

Adding new module securityAnalysis.py in the ACDC technique source code.

Add coding for visualizing the result on the reference of d3.js.

Prototype source code link:

<https://github.com/mcc12357/csci578>

Keywords for Access Control:

auth, authenticate, authenticator, authorization, authorize, identity, confidential, privacy, valid, validation, validity, account, access control, certificate, certificator, privilege

Keywords for Threat and Risk Assessment:

risk, threat, danger, val, evaluate, evaluator, safe, safety

References:

CSCI578 Course Lecture Slides Security and Trust

Wiki: <https://en.wikipedia.org/wiki/Secure_by_design>

<https://www.synopsys.com/blogs/software-security/software-risk-analysis/>

<https://d3js.org/>