# Scope

This assessment is based on a review of the 11,755 lines of Application Y services v2.0 source code received from X. This is a high-level summary using static source code scanners of both well known, SonarQube, and custom written scripts designed to support the DISA Application Development and Security STIG.

# Static Code Analysis

SonarQube 5.6.1 was used to perform a static code analysis with its default Java profile to determine findings based on industry-recognized best practices and patterns. These practices ensure a code base is written so that the compiled application is both secure and maintainable. The aggregation of these issues defines the technical debt of the application, which can be ignored, but will ultimately result in an unmaintainable code base that will only increase in difficulty and risk the longer it is left unaddressed.

**Blockers: 1**

* Single case of a FileInputStream that should be closed.

**Criticals:16**

* Primary concerns expressed as exception management within routines related to serving content/responses out, example: doGet(). Not a security concern but would make for cleaner code. Story created to address this in the future.
* Some false positives (MongoDB password) that are not of concern.

## Recommendation

The static code analysis does not reveal security concerns that would prevent deployment.

# STIG Assessment: Application Security and Development

**Anti-Pattern:** multiple references to “https” literal that could be collapsed to a single constant as well as multiple references to numeric values in REST API Services. Stories created and attached to refactor/address in the future.

## Recommendation

The STIG code analysis does not reveal security concerns that would prevent deployment.

# Engineering Assessment

* Stories will be registered in the backlog targeting potential refactoring that could improve the overall engineering stance of the project and reduce what is known as “Technical Debt”, however subject project (Some Services v2.0) will run and remain compliant if subject stories are not addressed.
* Apache Velocity Engine library released version 1.7 in November 2010 and shows no new development (http://velocity.apache.org/news.html).

# Questions

* None