Test Plan and Test Scripts

Security Violation Scanner for Email

Rowan University

Software Engineering I - Fall 2016

Team Eagle:

- *Mike Bayruns*
- *Christopher Deck*
- *Steve Leonetti*
- *Tom Miller*
- *Christopher Porch*
- *Dan Smith*

1 Introduction

1.1 Document Purpose

The Validation Protocol document contains the Validation Procedures and will conducted according to the corresponding Validation Plan.

1.2 Document Organization

The remainder of this document is organized as follows:

- Section 2 , Validation Strategy, contains the test levels and objectives, hardware and software required for validation, Resource and training requirements
- Section 3 contains a summary of the completed System Testing
- Section 4, Validation Methodology, explains the scope of validation, the organization and sequencing of validation procedures
- Section 5, Validation Procedures, contains the actual validation procedures

1.3 Project Background

This document contains the Validation Protocol to the Security Violation Scanner for email, designed for ASRC Federal Mission Solutions by a team of Rowan University Software Engineering students. The Security Violation Scanner for email aims to warn employees of potential confidentiality breaches in their emails before they are sent to prevent cases of data leaks.

1.4 Related Documents

Document Name	Version	Author
Project Initiation Document	1.0	Dan Smith
Requirements Document	1.0	Steve Leonetti
Design Document	2.0	Chris Deck
Validation Plan & Traceability Matrix	1.0	Mike Bayruns
Implementation Plan, incl. Change Management, Training	1.0	Tom Miller

2 Validation Strategy

2.1 Testing Levels & Objectives

2.1.1 Objectives

This document exists to ensure that adequate testing is performed in order to ensure:

- The Scanner's email analysis works as intended
- The Scanners database access works as intended
- Testing considers the use of different Operating Systems

2.1.2 Testing Levels

The levels of testing being done can be split into three types. These are Unit tests, Administrative tests, and Non-Administrative tests. Each level tests different functionality.

2.1.2.1 Unit

The Unit Script tests the JUnit tests that are built into the code, to test each class in the Violation Scanner and their functionality at the level of the source code itself.

2.1.2.2 Non-Administrative

The Administrative Scripts test functionality that an average user would have access to, including:

- Email entry
- Email scoring

2.1.2.3 Administrative

The Administrative Scripts test functionality that a user with some level of security clearance would have access to, including:

- Login
- Adding confidential words or phrases to the database
- File selection

2.2 System Requirements

This section details the system configurations required for testing in the Validation Environment, which should emulate the production environment as closely as possible.

2.2.1 Hardware

The testing will occur on standard desktop PCs with a connection to the internal or external server containing the database.

2.2.2 Software

The testing will take place on three separate desktops:

- Desktop 1: running Microsoft Windows 10
- Desktop 2: running Linux Mint
- Desktop 3: running Mac OS Sierra

In order to run the Unit Test Script, Apache Maven 3.0 or higher will need to be installed, as well as JDK 1.7 or above.

2.2.3 Source Code

The primary source code programming language used for the Security Violation Scanner for Email is Java 7.

2.2.4 Libraries/Directories

All External Libraries are organized using Apache Maven, and more details are available in /sweng-eagles/pom.xml, but a general list is:

- JUnit : 4.12
- Hamcrest, core : 1.3
- MySQL connector : 6.0.4
- Log4j : 2.7
- Apache Lucene, core & analyzers-common packages : 6.2.1
- JBCrypt : 0.3

2.2.5 Migration of Software and Data

Validation Environment test data will be included in the /sweng-eagles/test/ directory, or will be entered manually, as the test system has known “dummy” confidential words and phrases that can be used to validate scoring functionality.

3 System Test Evaluation

These System Tests were designed by the Development Team, Scrum Master, and Product Owner, based upon communication with the client and the clients specifications. More information is available in the Validation Plan for this project.

4 Validation Methodology

4.1 General Information

The approach to developing the following Test Scripts and this Test Plan was to approximate general usage of this program in a way that concisely touches upon each feature, as well as Unit Testing of the expected functionality of each class in this program.

4.2 Organization of Validation procedures

There is no preferred order of validation procedures, as they have been designed to be run in any order. This is due to the fact that there is no real dependencies or I/O between the functionalities tested in each script.

4.3 Validation Data Files

The only validation data files included for testing are .CSV files. These have been set up to test the .CSV file importing functionality, and test for appropriate parsing and error handling of such files before entry into the database.

4.4 Test Case/ Requirements Traceability

See Traceability Matrix

5 Validation Procedures

5.1 Validation Cases

The three validation cases are Unit Testing, Administrative usage, and Non-Administrative usage

5.2 Validation Scripts for Test Case SVSE

See 5.3.1.1 Unit Test Script, 5.3.1.2 Non-Administrative Script, 5.3.1.3 Administrative Script

5.3 Test Summary Report

5.3.1 SVSE

These Test cases test the Security Violation Scanner for Email using a network connection and Apache Maven.

- JUnit tests
- Logging in
- Email scoring
- Word entry
- Phrase entry
- Csv file selection

5.3.1.1 Unit Test Script

ID	Step	Input or Prerequisites	Procedure	Expected Result	Actual Result	Date	Tester's Initials	Pass/Fail
1.	1	Command line navigation, Apache	If on Windows, open cmd window, if on Linux/Mac open a Terminal window	Cmd or Terminal window opens				

	2	Maven installed	Navigate to the directory /sweng-eagles/ by entering the `'cd'` command into the cmd/Terminal window like so: `'cd *'`, substituting `'*'` for the file path the directory is located in.	Cmd or Terminal moves to directory				
	3	Internet Connection	Run command: `'mvn test'`	Maven downloads required libraries and runs JUnit tests. Output should say Tests run: 29, Failures: 0, Errors: 0, Skipped: 0, followed by BUILD SUCCESS.				

5.3.1.2 Non-Administrative Script

Test	Step	Input or Prerequisites	Procedure	Expected Result	Actual Result	Date	Tester's Initials	Pass/Fail
1.	1		Open sweng-eagles folder, then open target	Target folder opens				

	2		Double-click on sweng-eagles-3.0.jar	Email Text Input window opens				
2.	1: Empty text field		Click ‘Evaluate Email’	In the bottom right corner of the window, the word ‘green’ should appear.				
3.	1 : Scoring		Click in the text field and enter the word ‘clean’, then click ‘Evaluate Email”.	In the bottom right corner of the window, the word ‘green’ should appear.				
	2		Click in the text field and enter the word ‘word’, then click ‘Evaluate Email” again.	In the bottom right corner of the window, the word ‘yellow’ should replace the word ‘green’.				
	3		Click in the text field and enter the phrase ‘hello there’, then click ‘Evaluate Email” again.	In the bottom right corner of the window, the word ‘red’ should replace the word ‘yellow’.				

4.	1 : Failed login		Click ‘Import Terms’	Login window appears				
	2		Enter ‘admin’ into the username field, and ‘12345’ into the password field	Login unsuccessful				
	3		Click ‘Cancel’	Login window closes				
5.	1		Click Close button in Title Bar	Window closes				

5.3.1.3 Administrative User Script

Test	Step	Input or Prerequisites	Procedure	Expected Result	Actual Result	Date	Tester’s Initials	Pass/Fail
1.	1		Open sweng-eagles folder, then open targe	Target folder opens				

	2		Double-click on sweng-eagles-3.0 .jar	Email Text Input window opens				
2.	1		Click ‘Import Terms’	Login window opens.				
3.	1 : Login	Internet connection, or local network connection to server	Enter ‘admin’ as the username, and ‘asrcSw3ng’ as the password, then click ‘Login’	Username and password work, Database Input window opens.				
4	1		Click ‘Upload File’	A file selection window opens.				
	2		Navigate through the file selector to the sweng-eagles folder, then select test0.csv and press ‘Open’	Text to right of Upload file button should appear and say’ Processed test0.csv				
	3		Click ‘Upload File’	A file selection window appears.				
	4		Navigate through the file selector to the sweng-eagles folder, then select test1.csv and press ‘Open’	Text to right of Upload file button should appear and say ’Processed test1.csv				

	5		Click ‘Upload File’	A file selection window appears				
	6		Navigate through the file selector to the sweng-eagles folder, then select test3.csv and press ‘Open’	Text to right of Upload file button should appear and say ’Processed test3.csv’				
5	1: Cancel selection		Click ‘Upload File’	A file selection window appears				
	2		Click the ‘X’ in the title bar, or click cancel	Text to right of Upload file button should appear and say ‘Command cancelled’				
6	1: File detection		Click ‘Upload file’ and use the file selector to select any file that is not a csv file	Text to right of Upload file button should appear and say ‘Not a .csv file’				
7.	1:Word/ phrase entry		Enter a word into the word text field and a phrase into the phrase text field, and press submit	Text to right of Upload file button should appear and say ‘Processing complete’				
	2		Repeat previous	Text to right of				

			step with both' # Dependent?' radio buttons checked	Upload file button should appear and say 'Processing complete'				
	3		Repeat previous step, but with the number '0' entered in the 'Enter probability' text fields	Text to right of Upload file button should appear and say 'Processing complete'				
	4		Enter a word into the 'Words' text field, and check the 'Synonyms?' radio button, then press 'Submit'	Text to right of Upload file button should appear and say 'Processing complete'				
8	1: Empty text		Empty all text fields then press 'Submit'	Text to right of Upload file button should appear and say 'Processing complete'				