EmailScanner: Requirements Document (version 1.0)

Project: EmailScanner
Date(s): 10/16/16
Prepared by: Steve Leonetti

1. Introduction

This document contains the system requirements for ***EmailScanner***. These requirements have been derived from several sources, including ***MSE***.

1.1 Purpose of This Document

This document is intended to guide development of ***EmailScanner***.

1.2 Overview of the Requirements Document

The system must scan emails and reply with a score of the confidentiality of the email. If the email contains no sensitive information, the email will be simply release. Otherwise, if there is a report of possible confidential information residing in the email, the system will let them know. The user would be able to override to his/her discretion, although a report will be sent to a security administrator.
Administration will have the power to add new terms using file importation or the Graphic User Interface.

2. General Description

2.1 User Characteristics

MSE's workers will be expected to use this system to help them resist sending sensitive information via email. They will be motivated to use this product, because the administration expects sensitive information to be kept away from emails. They should require no more special skills to use this system: it will be as easy as trying to send an email and simply awaiting the A-OK of the system to continue sending the message.

2.2 General Constraints

We expect to be able to have our system as a plug-in for our client's usage, otherwise the client would need to manually input the text of the email to this process.

2.3 Assumptions and Dependencies

This system could depend on its acceptance with the Outlook email client. We expect the need of a Database to store our quarantined words and phrases.

3. Specific Requirements

This section of the document lists specific requirements for ***EmailScanner***. Requirements are divided into the following sections:

1. User requirements. These are requirements written from the point of view of end users, usually expressed in narrative form.
2. System requirements. These are detailed specifications describing the functions the system must be capable of doing.

3.1 User Requirements

Functional:

1. *Allow the user to override the email denial and send the email out, regardless.*
2. *Provide the administration a user interface to accept new term(s) and phrase(s) to add to the database of confidential terms.*
3. *Allow the security administrator to import, via CSV(Comma Separated Variable) file, new terms of confidentiality into the database.*

Non-functional:

1. *The user must be able to have emails scanned from their email client.*

3.2 System Requirements

Functional:

1. *The system must take a segment at a time of the email and check if the database contains that term or phrase and return a score of the term if applicable.*
2. *The system must score the total email body based off of the highest individual score.*
3. *The system must detect repeated patterns in the email body to recalculate probability of the email being confidential.*
4. *The system must locate the body of the email file and pull it from the email; this will include emails containing HTML formatting or Base-64.*
5. *The system must locate and decode PDF and Word files attached to the email.*
6. *The system must link synonyms together in the database.*
7. *The system must be able to load an email into the Scanner to process its score.*
8. *The system must take any confidential term or phrase in the email and check its context to possibly raise or lower the threat level.*
9. *The system must calculate a term's occurrence probability based off of common English speech.*

Non-functional:

1. *The system cannot store confidential terms and phrases in plain text, so that no unauthorized user can easily learn the terms.*
2. *The system cannot story every iteration of the same word. There must be one root word stored to reduce the problem space.*