20SGX– Spam Detection over Encrypted Emails

by

Brian Ciepichal, Reid Efford, & Jacob Welly

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

Reid Efford, Brian Ciepichal & Jacob Welly — 4/29/2019
Date

Bogdan Vykhovanyuk & Boyang Wang — 4/29/2019
Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

May 2019

TABLE OF CONTENTS

Section	**Page**
Abstract	1
1. Problem Statement	1-1
1.1 Introduction	1-1
1.2 Project Description	1-1
1.3 Problem	1-2
1.4 Solution	1-3
1.5 User Profile	1-4
1.5.1 Project Title	1-4
1.5.2 Potential Users	1-4
1.5.3 Software and Interface Experience	1-5
1.5.4 Experience with Similar Applications	1-5
1.5.5 Task Experience	1-5
1.5.6 Frequency of Use	1-6
1.5.7 Key Interface Design Requirements That the Profile Suggests	1-6
1.6 User Case Diagram	1-7
2. Project Management	2-1
2.1 Budget	2-1
2.2 Objectives/Deliverables	2-2
2.3 Project Schedule	2-3
3. Technical Elements	3-1
3.1 *Network (Hardware / Infrastructure)*	3-2
3.2 Application	3-3
4. Testing	
4.1 Overview	4-1
4.2 Scope	4-1
4.3 Objective	4-1
4.4 Logging Testing and Reporting	4-2
4.4.1 Preconditions and Postconditions	4-2
4.4.2 Testing Procedures	4-2
4.4.3 Pass/Fail Conditions	4-3
4.5 Knowledge Gained	4-4
5. Conclusion	5-1
5.1 Fall Semester 2017	5-1
5.2 Spring Semester 2018	5-1
Appendix A. Additional Info (Code, Tech Info, Screen Shots, Poster)	a
Appendix B. References	b

List of Illustrations

TABLES

Item		Page
Table 1.	Project Budget	2-1
Table 2.	Project Objectives/Deliverables Due Dates	2-2
Table 3.	Initial Test Plan	4-4

FIGURES

Item		Page
Figure 1.	Use Case Diagram	2-1
Figure 2.	Fall Project Schedule and Gantt Chart	2-3
Figure 3.	Spring Project Schedule and Gantt Chart	2-4
Figure 4.	Deliverables List	3-1
Figure 5.	Network Diagram	3-1
Figure 6.	Application Diagram	3-2

ACRONYMS AND ABBREVIATIONS

SGX Software Guard Extensions

ABSTRACT

Spam Detection over Encrypted Emails with Intel SGX Emails play a pivotal role in business and everyday communication. There are millions of email messages being sent each day that contain incredibly sensitive information. This sensitive data can be easily parsed by passive attackers on an email server. Leveraging Intel's SGX we can mitigate this problem by using enclaves. These are private regions of memory that are protected from processes running at higher privilege levels. Decrypting emails using SGX on an Encrypted mail server makes it much more difficult for passive attackers to get any useful data. On top of that this solution takes into account that sender reputation filters are not very effective since it is quite easy for an email address to be compromised. Decrypting emails via Intel SGX improves the general security of Emails on servers where it is running.

1. PROBLEM STATEMENT

1.1 Introduction

Emails are one of the most preferred communication methods currently in use today. Companies like Google, Microsoft and Yahoo put some effort in securing emails, but overall the job they do is not enough. Using Intel's Software Guard Extensions 20SGX will increase email security with better spam/phishing detection and encryption methods.

1.2 Project Description

Applying end-to-end encryption can successfully protect the privacy of users' emails, but it inevitably introduces new challenges to spam detection since the content of emails are encrypted on an email server. The objective of this senior design is to develop a secure spam detection scheme by leveraging secure enclaves (e.g., Intel SGX). Specifically, encrypted emails are only decrypted inside an enclave (i.e., a private region of memory) on an email server, and spam/phishing detection will be evaluated inside the enclave to minimize privacy leakage to potential hackers or insider attackers on an email server.

1.3 Problem

Emails are a key aspect of business and personal communication. According to the leading tech info website Comparitech more than 95% of people could not identify phishing emails. Due to this fact, attackers can easily get sensitive data from many users. A way around this is to apply end-to-end encryption and decrypt the emails using Intel’s SGX while simultaneously conducting content analysis on the emails. This allows for the encryption and content analysis to be done in a completely secure environment.

1.4 Solution

20SGX will be a C++ application that protects email users. It uses Intel Software Guard Extensions to create Enclaves in memory. These Enclaves are private regions of memory that give unparalleled protection to any code that is ran on them. Leveraging the enclaves with encryption create an email environment that is much more secure, even when the email server is compromised.

20SGX will be implemented on a desired email server that requires more protection. Once this is running on the email server the users on that email server will not notice anything different. They will be able to send and receive emails just as they did before, but now they will have an extra layer of security. The content of the user's emails will be much more secure and passive attackers will have a much harder time getting useful info from the emails.

20SGX will also add basic content analysis to the email server. The emails will be vetted and flagged as either safe or unsafe. Safe emails will be added to inboxes as usual while unsafe emails will trigger a warning email to the user. This content analysis will lessen the chance that someone opens and consumes dangerous malicious emails.

1.5 User Profile

We have determined that the End User is a user that wants a more secure email server. The most common end user would be people in businesses looking to further lock down and secure their email server. 20SGX will greatly increase security for anyone interfacing with emails on an email server where 20SGX is installed due to the security of enclaves.

1.5.1 Project Title

20SGX- Spam Detection over Encrypted Emails.

1.5.2 Potential Users

- Anyone looking to increase email security on an Email Server.
- The most common user will be companies that do not trust current email security and are looking for something more robust.

.

1.5.3 Software Experience

The project will be aimed towards anybody that will be sending emails on their network. The target of users can be in the thousands, many of the clients would be businesses wanting to have a more secure email server. Users will not be directly interacting with 20SGX it will be doing everything in the background. Their experience with the email server will be unchanged.

1.5.4 Experience with Similar Applications

- Transport Layer Security
- OpenPGP

1.5.5 Task Experience

- Install 20SGX on email server; 20SGX encrypts/decrypts emails and does content analysis without user input
- Users' experience with the email server will be dependent on the email server not 20SGX

1.5.6 Frequency of Use

20SGX will be intended to be used once it is installed it will be used every time an email is sent to the email server.

1.5.7 Key Interface Design Requirements That the Profile Suggests

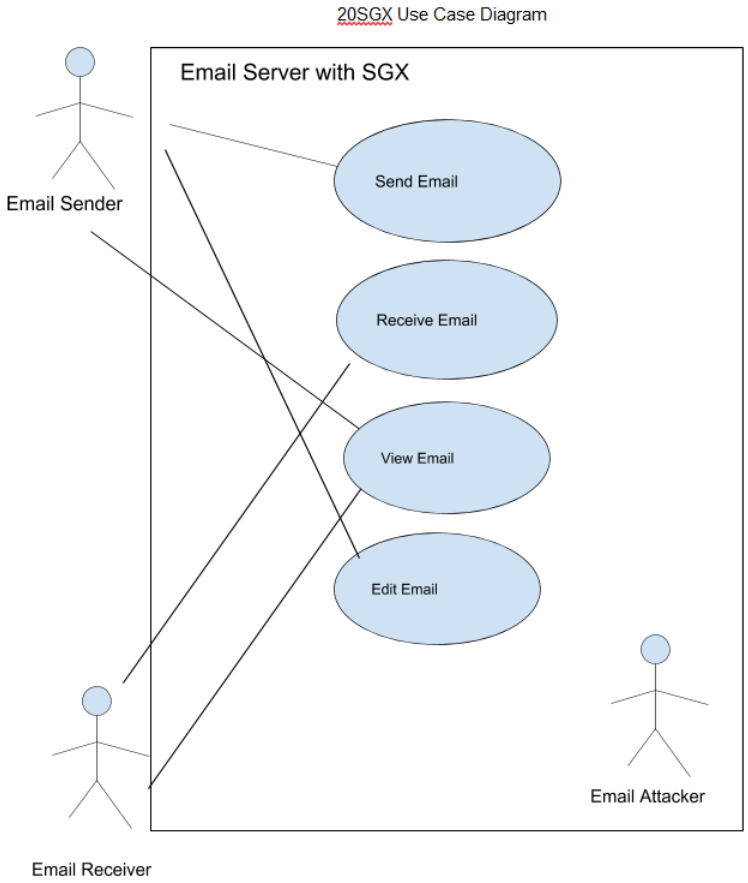
20SGX will run in the background of the email server. It will be used whenever an email is received on the email server.

- Easy to implement
- Minimal maintenance

1.6 USE CASE DIAGRAM

Figure 1: Use Case Diagram presents the project use case diagram. 20SGX use cases for users will be the same for users on the email server. There are two use cases. The users will be able to send and receive emails as usual, but if there is an attacker on the email server it will be much more difficult for them to get any useful data on the server. This is due to the secure code running on the server encrypting emails as well as running content analysis.

Figure 1: Use Case Diagram



2. PROJECT MANAGEMENT

2.1 Budget

Table 2: Project Budget presents the project budget. We are trying to cut costs in every way possible to reduce the budget.

Table 2: Project Budget

ITEM	UNIT #	COST	TOTAL
Labor	500 Hours	$20	$10,000
Visual Studio Professional Edition	3 Licenses	$539	$1617
SGX Enabled Computer	1	$800	$800
TOTAL			$12,417

2.2 Objectives/Deliverables

The project deliverables and deadlines are presented in Table 2: Project

Objectives/Deliverables Due Dates

Table 2: Project Objectives/Deliverables Due Dates

DELIVERABLES				
FALL OF 2017				
Enable SGX	10/27/2018		Develop Encryption/Decryption on SGX	11/10/2018
Create Enclave	11/3/2018			
SPRING OF 2018				
Implement Email Server	1/25/19		Implement Machine Learning Algorithm	3/5/19
Machine Learning Algorithm	2/28/18		Finish Testing	3/31/18

2.3 Project Schedule

Figure 2: Fall Project Schedule and Gantt Chart is our Fall project schedule.

Figure 2. Fall Project Schedule Gantt Chart

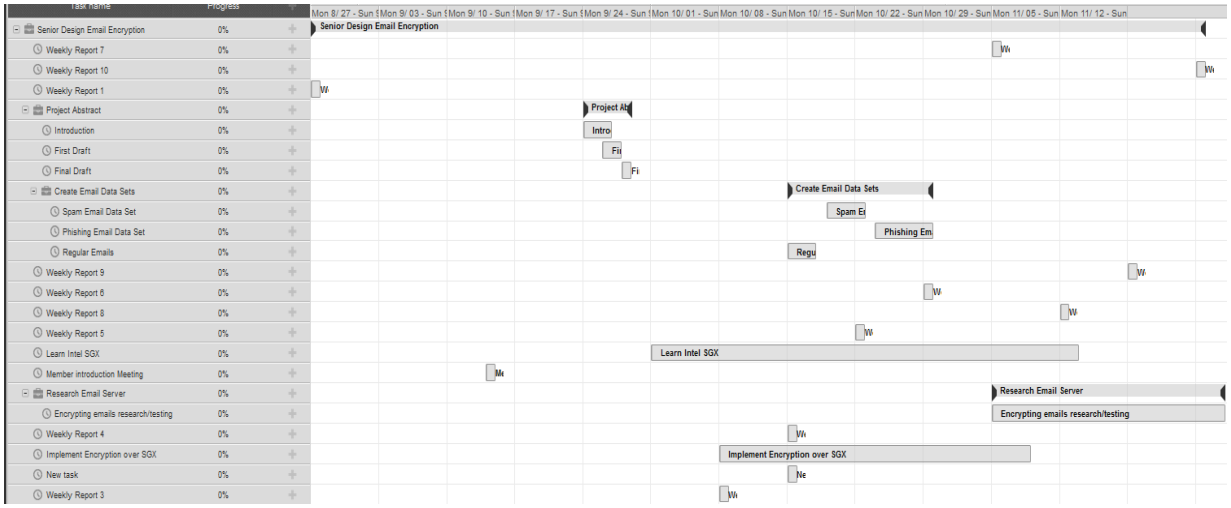


Figure 3: Spring Project Schedule and Gantt Chart is our Spring project schedule.

Figure 3. Spring Project Schedule Gantt Chart

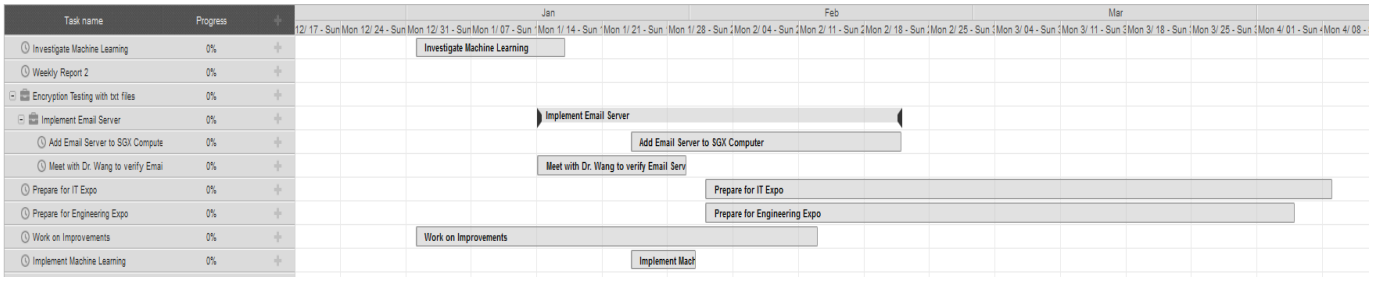


Figure 4: Deliverable Listing is a list of our project schedule.

Figure 4: Deliverable Listing

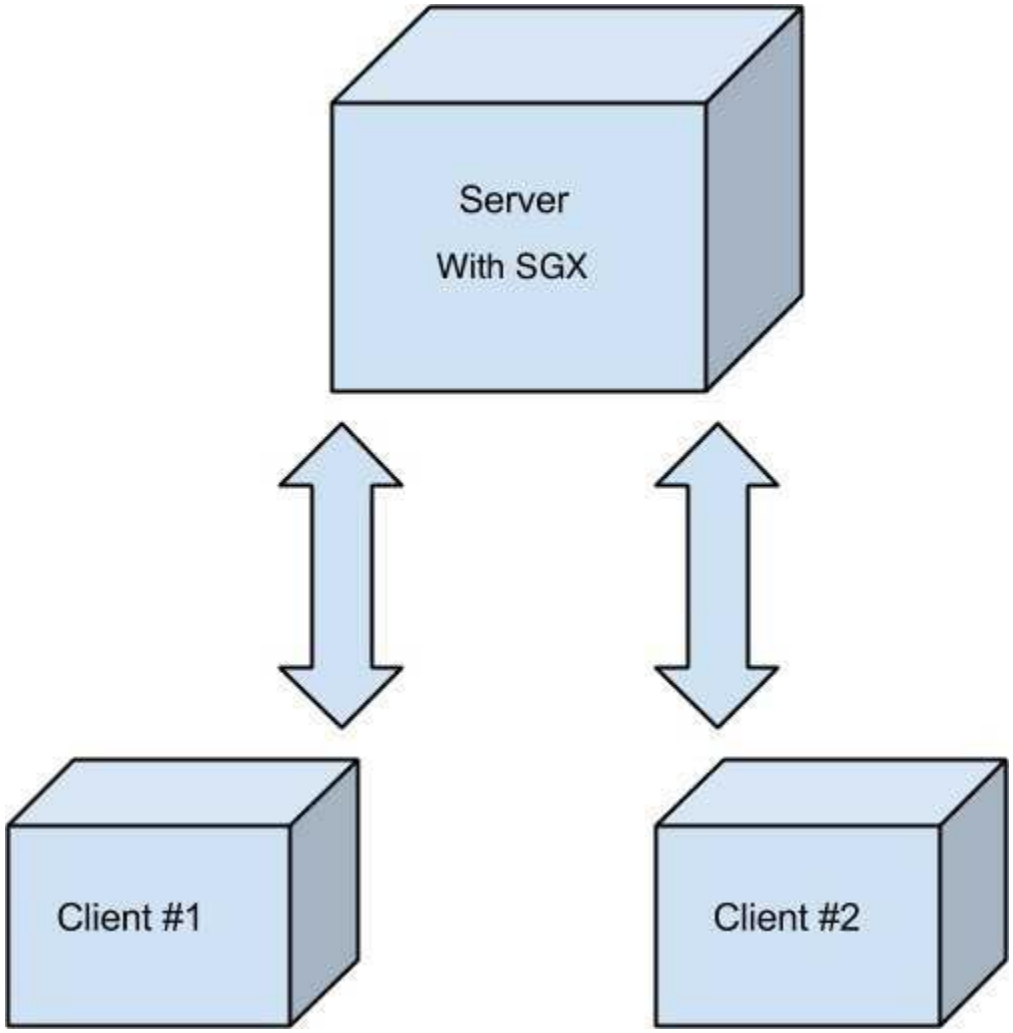
Task	Date Start	Date End
Member Introduction	9/14/18	9/15/18
Learn Intel SGX	10/1/18	11/14/18
Implement Encryption over SGX	10/8/18	11/9/18
Create Email Data Sets	10/15/18	10/30/18
Team Finalization	10/18/18	10/20/18
Research Email Server	11/5/18	11/29/18
Encrypting Emails Research	11/5/18	11/29/18
Discuss Improvements	12/3/18	12/7/18
Work on Improvements	1/1/19	2/13/19
Investigate Machine Learning	1/1/19	1/17/19
Implement Email Server	1/14/19	2/22/19
Implement Machine Learning	2/1/19	3/5/19

3. TECHNICAL ELEMENTS

3.1 Network

The application will be running on an email server. Depending on how the users email server is setup on a network the diagram could change. We are assuming that they have a simple client server network setup. Where each server can have multiple clients. 20SGX will be running on the server and will run code on enclaves on every email that is sent to the server.

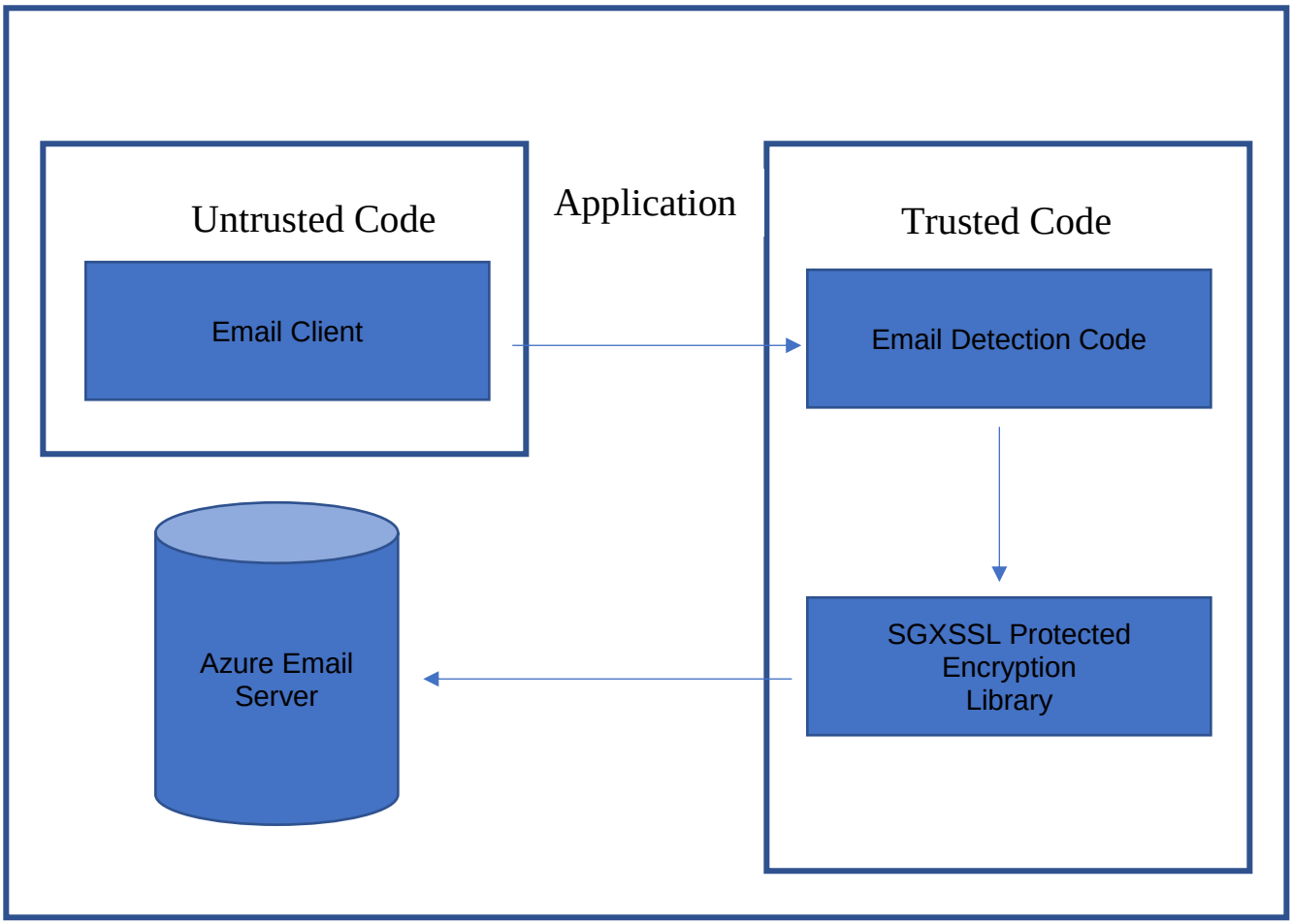
Figure 5: Network Diagram



3.2 Application

Our application will be built using C++ and Intel's SGX SDK. SGXSSL (OpenSSL) is used for encryption and decryption as well as the generation of the public and private keys. Our application is purely made for backend use as there is no front end for users to interface with. On top of that we will be using a dataset of safe and unsafe emails to allow 20SGX to determine which emails to flag as unsafe.

Figure 6: Application Diagram



4. TESTING

4.1 Overview

This section will explain the testing methodology for 20SGX and should be used as a guide for proper testing techniques. The following individuals should use this section:

- Developers
- Project Managers
- Q & A
- Cyber Security Workers

4.2 Scope

The scope of testing is to test the manner of which 20SGX handles email encryption and content analysis on an SGX enabled email server. In depth testing of the email client will not be done. The testing will be laid out based upon the objectives of 20SGX.

4.3 Objective

The objective of testing is to confirm that 20SGX is working as intended, so that users of 20SGX have the smoothest experience possible. The tests were created with the intention to either run them together or alone. The tests will be done by the 20SGX team.

4.4 Logging Test and Reporting

All the bugs found during testing will be documented by the Individual that found it. The problem code will then be analyzed by the developer who created it. They will then decide the severity of the issue and if the bug reported will need to become a new feature or just a small code change.

4.4.1 Preconditions and Postconditions

Preconditions:

~ Initiate Enclave creation

~ Verify Encryption Scripts

~ Content Detection Test

Postconditions:

~ Confirmation of test completion

~ Bugs are documented and organized

~ Bugs are analyzed and rectified

4.4.2 Testing Procedures

Steps for testing:

- Create a test planning document including all test cases and/or scenarios
- Document bugs clearly with steps to reproduce inside the test planning document

The following tests are how each test will be organized:

1. Enclave Test - This test will focus on confirming that the 20SGX enclave is successfully created.

2. Source Test - This test is to verify that the 20SGX application can run on an SGX enabled email server.

3.Performance Test - This test is to check the functionality of the 20SGX application and confirm that it is meeting the stated objectives.

Table 3: Initial Test plan

Tester	Procedure	Precondition	Postcondition	Result
Jacob	Source	Run application on AWS server	Application Fails	Fail
Brian	Source	Run app on Azure server	Application builds	Pass
Reid	Enclave	Create enclave on Azure Server	Enclave created	Pass
Jacob	Performance	Receive spam email	Spam email is flagged	Pass

4.4.3 Pass/Fail Conditions

20SGX needs to pass a test as stated by the test document. Failed tests will be documented within the test plan document with detailed steps on how to reproduce the issue.

4.4 Knowledge Gained

Testing 20SGX has been very beneficial regarding the development of the application. The email server had to be reworked from Amazon AWS to Microsoft Azure due to compatibility issues.

5. CONCLUSION

5.1 Fall Semester 2017

The focus of this project during the fall has been taking a deep dive into Intel SGX. We began the semester by obtaining this project from Dr. Wang and working closely with him to deliver what was expected. We began the semester expecting to create a whole email server but realized quickly that we had some hardware issues that made that difficult. We instead decided to focus on a concept of SGX during the fall since a lot of our time was spent trying to find hardware compatible with SGX and our other tools. It has been exciting using this new architecture to create something that can help millions.

5.2 Spring Semester 2018

In the Spring Semester we are going to be focusing on finalizing our application. In the Fall we were not able to get everything with our project finished. We spent most of the time during this semester working on creating a working spam and phishing detection system within SGX. This semester we finished working on this aspect of the application and created the Spam Detection system on the Enclave. Our application now identifies possible spam and phishing emails. These emails are quarantined on the email server to protect users from the harmful emails, while retaining email functionality with non-harmful emails.

APPENDIX A. ADDITIONAL INFORMATION

1. Bo Vykhovanyuk, Advisor
2. Boyang Wang, Electircal Engineering Advisor

APPENDIX B. REFERENCES

"Email Security – Essential Guide." *ComputerWeekly.com*, www.computerweekly.com/feature/Email-security-Essential-Guide?src=itke stub.

Goldschmidt, Cassio. "Modern Phishing Campaigns And Effective Prevention." *Forbes*, Forbes Magazine, 2 Mar. 2018, www.forbes.com/sites/forbestechcouncil/2018/03/02/modern-phishing-campaigns-and-effective-prevention/#697f3cfa649d.