UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS 720: Computer and Information Security Project Documentation

Armand Maree 120 178 00
Charl Meyers XXXXXXXX
Marc Antel XXXXXXXX

Department of Computer Science, University of Pretoria

Contents

1 Introduction

Spam and malicious emails is a world wide problem wasting billions of dollars every year [1]. Since 90% of all email traffic is spam, a lot of network bandwidth, processing power, storage and electricity goes toward the processing and transportation of these emails [1]. Due to this large amount of unwanted email traffic workers waste time everyday filtering these out. Not only that, but malicious emails could cause massive financial loss to an individual and/or company. For these reasons there have been numerous studies done to identify and filter out these emails. One such approach is header based spam filtering [2]. This approach requires the analysis of large datasets in order to accurately classify emails [2].

2 Data Acquisition

On March 26, 2003 one of the most widely used email corpuses was published on the Internet by the Federal Energy Regulatory Commission (FERC) [3]. This was of course the Enron email corpus [4]. There were two reasons why this data was chosen for this project, the first is due to this data set's wide and successful use in previous research, and the second reason is due to the massive number of "real-world" emails that it provides.

3 Data Cleaning

There were multiple steps that played a role in the data cleaning process. The main purpose of this step was to place the data in a CSV format that could be used for further analysis. These steps will be discussed in the next sections.

3.1 Email Format

Due to previous research and individuals like Melinda Gervasio [4] the emails have been clean up drastically and reformatted to follow a more standard layout. The emails start out with all the headers and followed by the email body. These two are separated by a blank line. Since we are only interested in the email headers, we will focus on this section more extensively. The general header layout consisted of the header followed by a colon and then the header's value. Occasionally the value spans over multiple lines since a line is not allowed to be more than 998 characters long (although 78 characters is recommended) excluding the CRLF characters [5]. Any data after the blank line that separated the headers and the body was ignored.

3.2 Data Extraction

3.2.1 Header Acquisition

Each line of the header section of the email was scanned and split by the colon character based on the RFC specification for email layouts [5]. Any text before the colon is seen as the header key that follows the colon is seen as the header value. As already mentioned, there were cases where the header value wrapped across multiple lines. This is allowed as long as the new line starts with a white space character [5]. However, there were a few emails that did not start the new line with a white space character. These emails were flagged by the cleaning program and were corrected manually. Header values that spanned across multiple lines were combined into a single line.

The headers (with descriptions) that were found in the complete set of emails were:

- Received1 (fabricated header)
  Contained the address of the receiving server. Example: *by 184.168.221.41 with SMTP id hUFS1mAtWD54sRsA; Mon, 14 May 2001 16:39:00 -0700 (PDT).*
- Received2 (fabricated header)
  Contained the address of the sending server as observed by the receiving server. Example: *from enron.com (enron.com [184.168.221.41]) by mx.google.com with ESMTPS id IA15iqgM3n4tZP; Mon, 14 May 2001 16:39:00 -0700 (PDT) (version=TLS1_2 cipher=AES128-GCM-SHA256 bits=128/128); Mon, 14 May 2001 16:39:00 -0700 (PDT).*

- X-Mailer (fabricated header)
  This field was left blank in the emails but usually contain the email program that was used to compose the program [6].

- Message-ID
  The message ID of an email consists of a global unique ID to identify this email. It usually, although not in the case of the Enron emails, ends with the domain of the sending server. Example: *<18782981.1075855378110.JavaMail.evans@thyme>*.

- Date
  The time stamp the email was sent. This time stamp has a very specific format which is: Day of the week, Day Month Year Hour:Minute:Second +/-Time zone offset. Example: *Mon, 14 May 2001 16:39:00 -0700 (PDT)*

- From
  The sender email address. Example: *piet.pompies@enron.com*.

- To
  A list of one or more recipient email addresses separated by a comma. Example: *johan.kotze@enron.com, koos.opperman@enron.com*.

- Subject
  The subject of the email as specified by the sender.

- Mime-Version
  Multipurpose Internet Mail Extensions (MIME) is a header that indicates that the message has been formatted in accordance with the MIME format standard [7]. At the moment there is only one MIME version (version 1.0) [7] and there hasn't been a need to call for a new official version.

- Content-Type
  Also referred to as the media type [7], and is used to allow the received to choose the correct mechanism to display the email to the user. The default for this field is *text/plain; charset=us-ascii* [7].

- Content-Transfer-Encoding
  The primary purpose of this field is to allow the receiver to decode binary data that was converted to a text format. This field specifies how this data was encoded [7]. The default value is *7bit* [7].

- X-From
  As stated in [8] any header starting with an "X" is considered application specific and will not be overwritten by a future standard. This has since been deprecated [9] but only after the Enron emails have been sent.
  In our data set this field contained most often contained the sender's name and surname, their email address and/or other random information. Example: *Mass, Frans </O=ENRON/OU=NA/CN=RECIPIENTS/CN=FSAYRE>*.

- X-To
  Similar to the X-From header.

- X-cc
  Similar to the X-From header.

- X-bcc
  Similar to the X-From header.

- X-Folder
  This header appears to store the folder the email was placed in, in the user's email program.

- X-Origin
  Most often this field contained the sender's surname and firstname's initial. Example: *Mass-F*

- X-FileName
  The Note Storage Facility (NSF) or Outlook Personal Folders (PST) file name.

- Cc
  A list of one or more recipient email addresses of individuals that was added to the carbon copy list separated by a comma. Example: *johan.kotze@enron.com, koos.opperman@enron.com.*

- Bcc
  A list of one or more recipient email addresses of individuals that was added to the blind carbon copy list separated by a comma. Example: *johan.kotze@enron.com, koos.opperman@enron.com.*

3.2.2 Removed Emails

Complete removal of emails were left as a last resort in order to maintain the large numbers of emails. The only emails that were stripped completely were emails that did not contain a header value for the "To" header key. The reason for this is because these emails were seen as draft emails that were not sent out.

3.2.3 Data Preparation

Some of the fields were reformatted in order to provide a simpler data set which would in turn allow for a more successful data mining process. This includes the date field which was reformatted to the yyyy-MM-ddThh:mm:ss format. Missing headers were added with an empty value. Quotes and commas were stripped from the fields to allow CSV compatibility. For the Received1 and Received2 fields, only the IP addresses were extracted of the corresponding servers.

Some additional data was also gathered from the emails and added as additional fields. This includes:

- Message-ID-Server
  If the message ID contains a domain name then it is extracted and placed into this field.

- From-IP
  The actual IP address were requested for the sending email address's domain. Some domains' IP addresses could not be retrieved and in this case the domain name was placed in this field.

- File-size
  The size of the email in bytes.

3.2.4 Data Conversion

In order to allow the machine learning phase to be able to more easily mine the data, the text field were converted to numerical representation and a mapping file (for our reference) was maintained. This includes the following fields:

- Message-ID-Server
- Mime-Version
- Content-Type
- Content-Transfer-Encoding

Other fields had some aggregation performed in order to get a numerical value. Like the recipient fields (listed below) were converted the number of recipients in that fields. For instance, a "To" field that contained 3 recipient email addresses has it's value replaced by the number 3.

We thus maintained two copies of the data, and either could be used by any phase of the project. The first copy is the original string formatted data and the second is the numerical data.

4 Exploratory Data Analysis

In order to gain a more in depth understanding of how the data looks we performed some Exploratory Data Analysis (EDA). EDA is a method used in research to gain knowledge of the data that could in turn generate ideas

and provide a possible path of investigation to follow [10].

A script was written to extract data from the cleaned data file and draw some Excel graphs to provide a visual mechanism of interpreting the data. The following analysis was performed as part of the EDA:

- Number of emails sent per user (figure 1)
- Number of emails received per user (figure 2)
- Number of emails sent during each hour of the day (figure 3)
- What content types were used (figure 4)
- What MIME-Types were used (figure 5)
- What transfer encodings were used (figure 6)
- What domains were listed as part of the message ID (figure 7)
- Some descriptive statistics were also performed to indicate what the max, min and average number of emails per user was (figure 8)

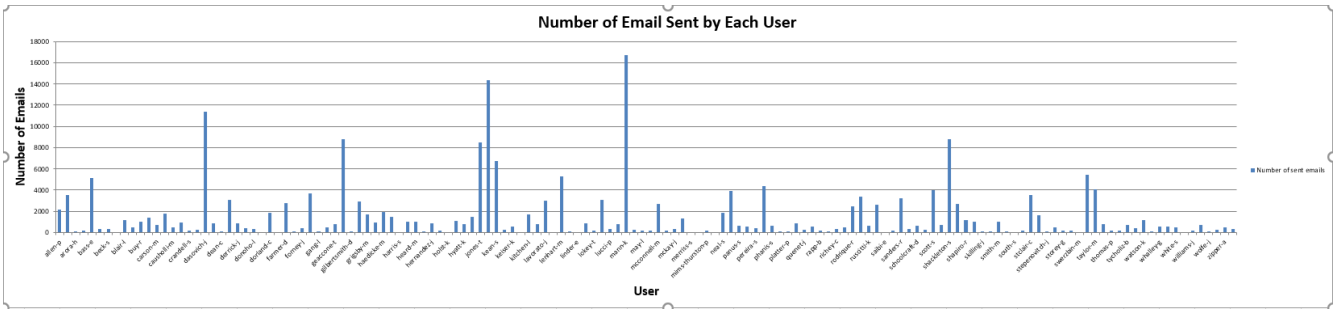

Figure 1: Number of emails sent per user.

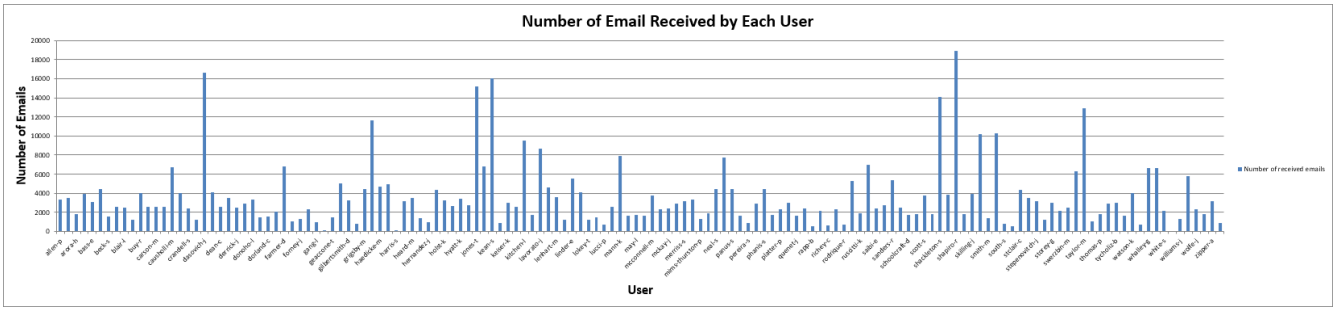

Figure 2: Number of emails received per user.

5 Attribute Identification

6 Feature Engineering

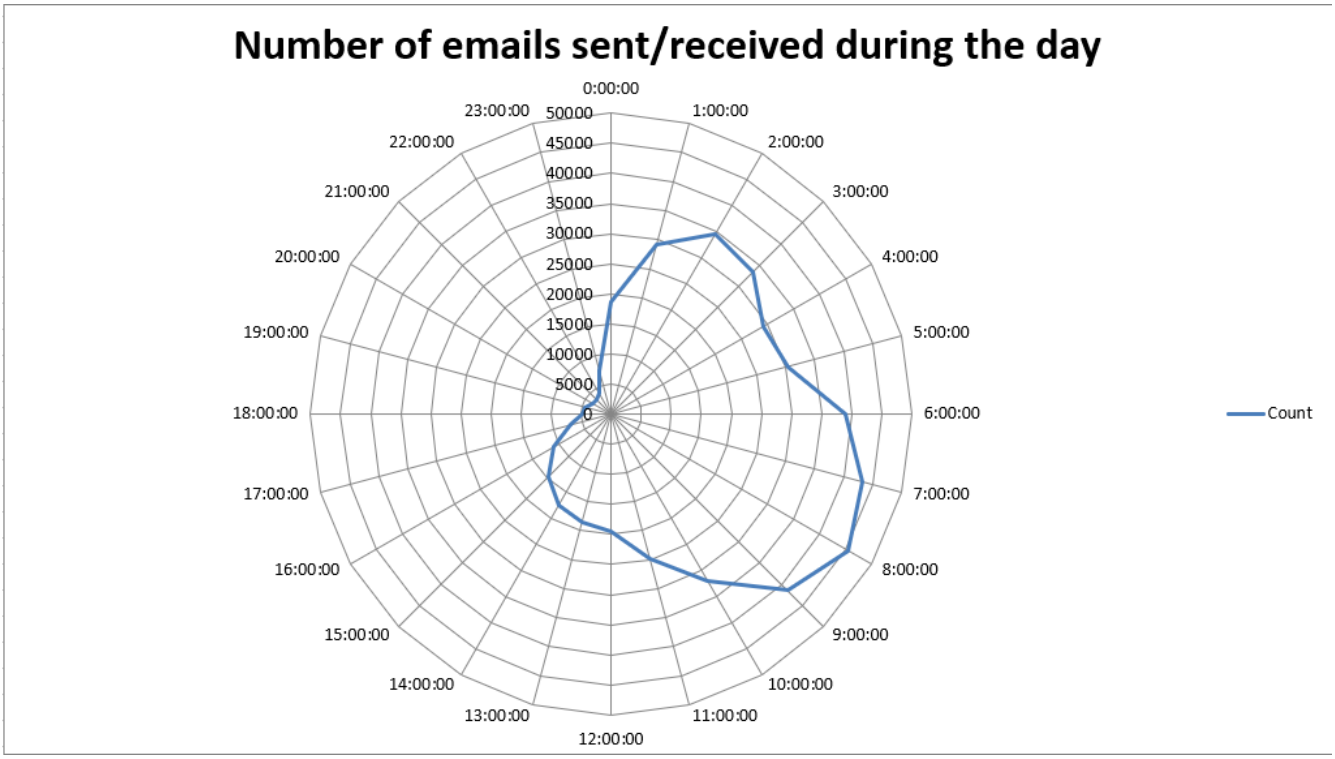

Figure 3: Time of the day emails were sent/received.

7 Using Machine Learning to Detect Abnormal Behavior

7.1 Technique 1

7.2 Technique 2

8 Criminal Profiling

9 Anonymization

References

[1] Daniel Castro. How to stop the billions wasted annually on email spam, July 2013.

[2] Omar Al-Jarrah, Ismail Khater, and Basheer Al-Duwairi. Identifying potentially useful email header features for email spam filtering. In *The Sixth International Conference on Digital Society (ICDS)*, volume 30, page 140, 2012.

[3] Jessica Leber. The immortal life of the enron e-mails, July 2013.

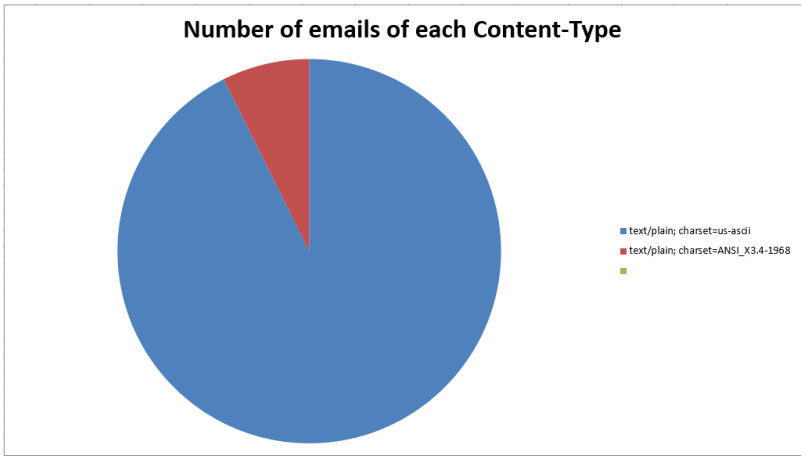

Figure 4: Content type used in emails.

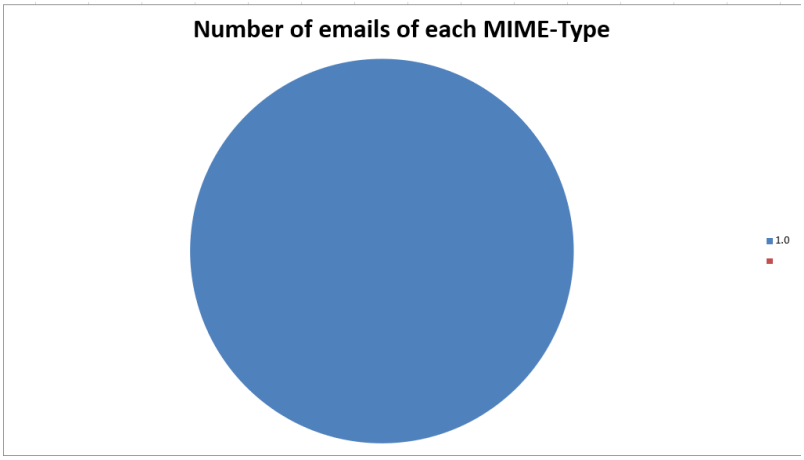

Figure 5: MIME type used in emails.

[4] Carnegie Mellon University. Enron Email Dataset, May 2015.

[5] Internet Engineering Task Force. Rfc2822: Internet message format, April 2001.

[6] Nicole Martinez. What is a x-mailer header?

[7] Internet Engineering Task Force. Rfc2045: Multipurpose internet mail extensions (mime) part one: Format of internet message bodies, November 1996.

[8] Internet Engineering Task Force. Rfc822: Standard for the format of arpa internet text messages, August 1982.

[9] Internet Engineering Task Force. Rfc6648: Deprecating the "x-" prefix and similar constructs in application protocols, June 2012.

[10] Victoria Cox. Exploratory data analysis. In *Translating Statistics to Make Decisions*, pages 47–74. Springer, 2017.

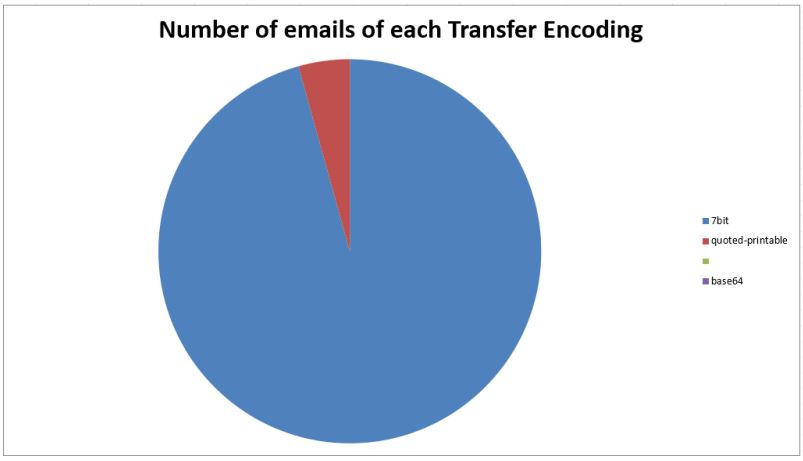

Figure 6: Transfer encoding used in emails.

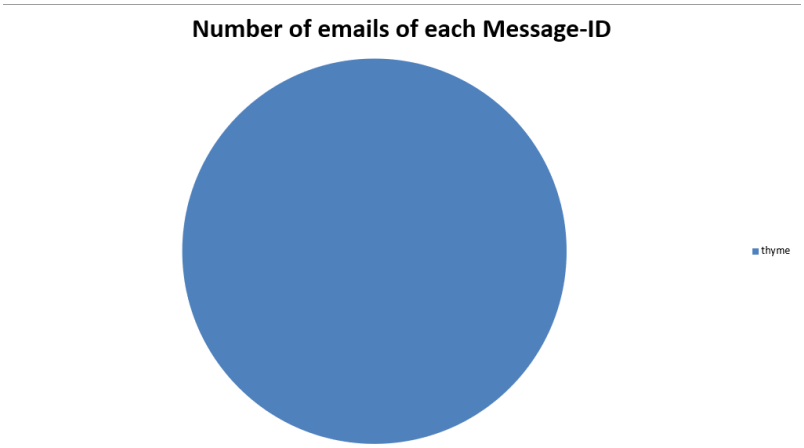

Figure 7: Message ID domain used in the emails.

	A	B
1	Max emails sent	16735
2	Min emails sent	0
3	Average emails sent	1470.267
4		
5	Max emails received	18917
6	Min emails received	51
7	Average emails received	1470.267

Figure 8: Descriptive statistics on sent/received emails.