The Spaminator

Mathew Hazel

Ira Nicks

John Healy

POS 355

Professor Long

January 12, 2004

The Spaminator

Unwanted solicitations via email have become an extremely large problem in today's society. No one wants to be interrupted during dinner time with an unwanted phone call solicitation or a knock on the door by a door to door salesman. There has even been a law passed by the government stating that telemarketers can not call someone if they are on the "No Call List". So how do you feel about unwanted email solicitations? Millions of unwanted emails are sent to electronic mailboxes worldwide each day costing the recipients of these emails valuable time sorting through the inbox to find needed information. This common occurrence slows down business resources and costs the companies receiving these emails valuable man hours. Our system solution to this problem is a database called "The Spaminator".

This database will contain all known email advertisements and unwanted solicitations. The database will then be connected to subscribing companies email server and all incoming email traffic will be compared to this database before being delivered to a client's inbox. All email matching the known spam email will be filtered out of the incoming mail. We are currently considering whether or not the incoming spam email has to match 100%, or if we will be looking for catch phrases and keywords.

Once we have to database information in place and online, we then plan to test the database by setting up an email server and connecting it to our database and start beta testing the system until it is perfected. When we have a success rate of about 80% success rate, we then plan to offer our service to a company for a limited time free of charge. This will allow our system exposure to the business world as well as a live test

run. From this live test run we hope to gain valuable feed back and word of mouth advertising.

The first individual requirement is the ability for the team to create a piece of software capable of determining if two pieces of text e-mail are at or above an 80% body content match. What this means is we will take an unknown e-mail and compare the text body of the e-mail to a database of known SPAM e-mail. When we produce a match of 80% or more of the known SPAM and the unknown e-mail we will then flag the unknown e-mail as SPAM. Once we have flagged the e-mail as SPAM we will be able to several things with this piece of e-mail. We will be able to produce reports for the people who subscribe to our service and say things similar to we stopped X amount of SPAM from getting to you.

On the hardware side of the project the plan is to buy the equipment and then set-up the internal network. The way this will be done is to pick a single room and install a razed floor for the data center. Once the floor is done the computers will be installed into the datacenter along with the switches and routers. Then cabling will be installed to connect everything together. Then once the T3 line is installed we will connect the network to the T3 line for connection to the Internet.

On the software side the plan is to get a program to be able to compare 2 e-mails. Then the next step will be to have the program compare the two e-mails and have a 100% match with a return of a Boolean yes for true. Then the next step will be to have the two e-mails compared with a match of 80% or more. Then once we have a working program we will create the code to allow the program to interact with a mail server and the database of known SPAM mail.

The designing of the Spaminator brought about major questions of how it should function. The potential size of the database presented problems that had to be solved by trade offs. The size of the system did not allow for a simple software solution that can be installed on a client's server. Due to the fact that a large amount of data had to be gathered and then compared to incoming email meant that a large amount of disk space needed to be allocated to store samples of unwanted email. The estimation of the database will be in the terabytes.

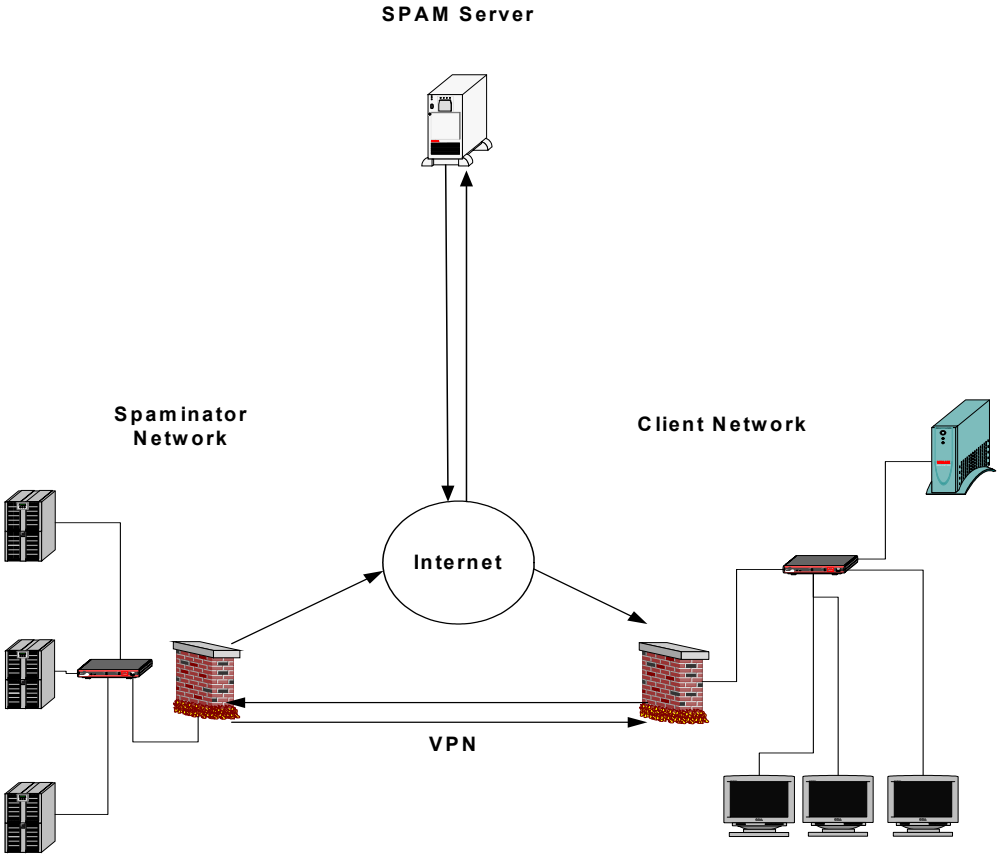
Another consideration that was made is whether or not to have a program similar to Norton on the server to route incoming mail on keywords. This will cut down on the amount of traffic the Spaminator network would have handle and minimize the need for a faster internet connection. The determination was made that this would not achieve that accuracy of filtration that we are looking for. In order to deliver the accuracy of 80% all traffic from the client's server would need to be routed to the Spmainator network for processing via a Virtual Private Network.

To achieve this we designed the Spaminator to be a server side solution. That means minimal on site work and our mainframe can be located miles away from our client's facilities.

For our software design we chose to build our own proprietary software to eliminate spam. As stated before, the server side solution allows us to not have to install software on every user's pc and we will not have to worry about licenses because we will own the technology. The Spaminator software has the capability to compare the e-mail's subject and body for known spam words or phrases to a database. If there is a match the

e-mail will not be allowed to go through. The Spaminator is customizable so words or phrases can be added and removed as needed.

Once a message is determined to be spam, it also adds the server from where the spam e-mail is being sent from and we can decide on whether or not the server needs to be added to our blacklist for spam servers. A message is also sent to the user notifying them a spam message was blocked with detailed information so the user can decide if the message was indeed spam and not a weekly newsletter that might get mistakenly flagged as spam.



There will be no hardware requirements for any of our clients except for an existing computer and network infrastructure that we can access. All hardware requirements will be on our side. We plan to run a mainframe that will be expandable depending on how many clients we have.

For the Spaminator project we had to choose a computer platform and an operating system to base our network and run our anti-spam software on. There were a few different platforms and operating systems we chose to evaluate that were technically capable of accomplishing the requirements and goals of our proposed system. In order to discern which platform and operating system would be the best fit for the situation, nine criteria were used for comparison with emphasis on criteria we consider most important.

- Ease of use
- Cost
- Efficient Use of Resources
- Portability
- Compatibility
- Scalability & Upgradeability
- Stability and Robustness
- Network and Connectivity
- Security

One of the computer platforms that were high in consideration is the CRAY X1 series computer platform. There is not a doubt that this system would successfully run

our anti-spam software and be the base for our network. CRAY promotes the X1 series as being efficient, scalable and high network interconnects.

The CRAY X1 series runs on the UNICOS/mp™ operating system which is UNIX variant. UNIX is one of the most secure Operating systems on the Internet today. The biggest security risks to the CRAY X1 computer system exist in physical system compromise. Physically securing the computer system from unauthorized users easily negates that problem.

UNICOS offers Standard Networking Support. The operating system provides NFS distributed network file system; IPv4 TCP/IP, including the standard UNIX socket API; Domain Name Service (DNS) and Network Information Service (NIS) as client; and Network Time Protocol (NTP).

After reviewing other technical specifications we deemed the CRAY X1 series computer platform to be an extreme overkill for our requirements. The Cray X1 system, designed to be the world's most powerful supercomputer product, features ultra-fast (12.8 gigaflops) individual processors, up to 819 gigaflops of peak computing power. For our spam filtering, that much power would be too much and there would be no means to fully utilize all of that power. There is also an issue with cooling the chips in the CRAY computers. Some methods suggest spraying the chips with nonconductive liquid called Flourinert and chilled water to cool the coolant to maintain a constant temperature.

The fact that CRAY advertises its X1 series for problems needing extreme performance such as drug discovery, energy and transportation modeling,

nanotechnology, severe storm forecasting and climate modeling, planning for natural pandemics and bioterrorism, spam filtering just does not warrant such a costly purchase.

The use of servers with the thought of eventually clustering was the first idea that was considered. This would be an economical idea to start the business since servers cost far less than buying a mainframe up front. They can also be acquired in a shorter time frame than could a mainframe. This would make running servers more economical and faster to acquire.

The problem with clustering servers is that we will be running a data farm in this project. It is faster processing wise to have data run through one mainframe than several PCs acting a server. Servers being ran in a cluster would also equal more down time in the future and would be a little harder to maintain than one mainframe. With the way we intend to market this service, the size of the cluster would become quite large if we land the contracts we are looking. If we can land an internet service provider, we could be in the use capacity of a mainframe from the start.

The first operating system that was first considered was Microsoft Windows Server 2003. The benefit of using this operating system is it is very user friendly and widely used. The problem with a widely used system is that there are more people on the internet with criminal intent working on ways to crack that system. Microsoft Windows has long had security issues. Microsoft is doing a good job of attempting to correct these issues, but they are not all ironed out yet.

The Digital Alpha Server is the correct size server for the company. With the cost of about 300,000 dollars and with room to expand it is the best product for the price.

With the Alpha Server we will be able to have scalability, stability, and the needed security.

With the Digital Alpha Server we will be using the VMS operating system (OS). One of the nice things about using the VMS OS is we will not have to be two concerned about viruses or with people hacking into the server. The reason for this is there has not been a virus unleashed onto the public for this OS in over 10 years. The second reason is the security of this OS is extremely strong. With the Alpha Server we will be able to have scalability because we will have room to grow. The VMS OS and the Alpha Server are extremely stable platforms. The only time this type of a computer goes down is when there is a hardware problem because something broke or power is lost to the computer. Because the Alpha Server will be running the VMS OS we will not have to worry two much about people hacking into the computer because it is a older type of OS and therefore the script kiddies would not be able to get into the computer and even if they did manage to brake into the server they would not know any of the commands to do anything to or in the server. The nature of the company the Digital Alpha Server will meet the best needs of the company and the Alpha Server will allow the company room to grow.

The plan for testing the Spaminator will consist of two parts; testing the proprietary software designed to filter the incoming email, and testing the load capacity of the network. The software designed for the system will be a form of artificial intelligence that will compare known undesirable email to incoming email. The initial software test will consist of gathering known undesirable email and using the Spaminator

software to filter the email past through the mainframe. The software itself will go through code and syntax checks to make sure it is functioning properly. Documentation will also be kept throughout not only the test phase, but the entire project. Once 80% accuracy is achieved, the software will be considered a success.

In order to test the software in a real live environment, the mainframe will be set up with the proprietary software on it and connected to a secondary email server which will act like a client server. The project team will access the internet and give out the email address of the dummy client server to as many public websites as possible. This will allow the Spaminator mainframe to start gathering undesirable email to put into its database. It will recognize undesirable emails by keywords and the presence of .jpeg and .bitmap files. The emails gathering in the database will also be followed closely by the project team to make sure the emails we are retrieving are considered undesirable. The project team will also be looking to see that emails not gathered and let through to the dummy email server are desirable email. The team will then make adjustments to the email gathering process as necessary.

Once the Spaminator software is performing to the 80% goal, the project team will then bring the mainframe, a router, and a T3 line online. The system will then be load tested to make sure the capacity of the system can stand up to a real life environment. The goal of this phase of testing is for the system to be able to handle 60000 emails per minute. Once this goal is achieved the system will go through a live test. The Spaminator's services will be advertised free of charge for a limited time to select companies interested in filtering their email. This part of the process will not only

give the system a real live test, it will also give the system credibility in the business world as the system of choice for filtering out undesirable email.

The next phase of the system will be the acceptance testing / changeover. In this phase we will offer the system to prospective companies for a period of 60 days free of charge. This will allow the system to prove itself in a real world environment. At the end of the 60 day trial period and evaluation of the system will be preformed to make sure the 80% accuracy level is being met. The project team will also check the software and hardware to determine if the scope of the network needs to be expanded or if the initial network will be able to handle the workload for the time being. A secondary database will also be created to log trouble tickets to further the uptime of the system. Every 15 days the system load will be evaluated. If the load of the system reaches 80%, the project team will expand the existing network and or bandwidth to accommodate growth of the system, the business, and profits.

On the mainframe we will have a daily, monthly, and yearly maintenance plan. The daily maintenance plan will be just checking the system log files and looking for any errors. The monthly maintenance plan will include a cycling of the power on the mainframe. A look at the overall performance and messages in the log files over the past month which includes looking for any specific error messages and a look for a performance trend. The installation of any new patches for the mainframe which have been released over the past month will be done at this time. The yearly maintenance will include cleaning out the dust in the mainframe and a performance check of the log files

over the past year. At this point you are looking for trends in performance rather then a specific error message.

Once the above mentioned goals are achieved we can have our on staff programmers begin working on perfecting the proprietary software. By increasing the efficiency of the program and catching more email, we hope to gain more clients. In time, we hope the Spaminator to be the spam filtration system of choice for companies and internet service providers around the world.